Examination of RTCA/DO-198 Position Reconstruction Algorithms for Area Navigation with the Microwave Landing System
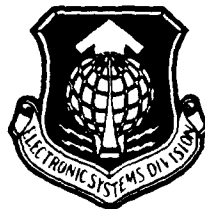
By

J. W. Hall
P. M. Hatzis
F. D. Powell

June 1990

Prepared for

Program Director for
Communications and Airspace Management Systems Program Office
Electronic Systems Division
Air Force Systems Command
United States Air Force

Hanscom Air Force Base, Massachusetts

AD-A224 804

REVIEW AND APPROVAL

This technic　eport has been reviewed and is approved for publication.

TRACY BOMBARD, 1LT, USAF
MMLSA Program Manager

DAVE KENYON, GM-14
MLS Program Manager

FOR THE COMMANDER

DAVID A. HERRELKO, COL, USAF
Program Director
Comm. and Airspace Management SPO

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION	1b. RESTRICTIVE MARKINGS
Unclassified	

2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	Approved for public release; distribution unlimited.

4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
MTR-10766 ESD-TR-90-308	

6a. NAME OF PERFORMING ORGANIZATION	6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION
The MITRE Corporation		

6c. ADDRESS (City, State, and ZIP Code)	7b. ADDRESS (City, State, and ZIP Code)
Burlington Road Bedford, MA 01730	

8a. NAME OF FUNDING/SPONSORING ORGANIZATION	8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
Program Director (continued)	ESD/TCVM	F19628-89-C-0001

8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS			
	PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.
Electronic Systems Division, AFSC Hanscom AFB, MA 01731-5000		5420		

11. TITLE (Include Security Classification)

Examination of RTCA/DO-198 Position Reconstruction Algorithms for Area Navigation with the Microwave Landing System

12. PERSONAL AUTHOR(S)

Hall, John W., Hatzis, Patricia M., Powell, Frederic D.

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM _____ TO _____	1990 June	59

16. SUPPLEMENTARY NOTATION

17.	COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Computed Centerline Approach, Newton-Raphson
			Gauss-Seidel, Position Reconstruction
			Microwave Landing System (MLS), Algorithm

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Use of the Microwave Landing System (MLS) for area navigation and computed centerline approach requires that the MLS avionics determine the aircraft position in Cartesian coordinates relative to the runway centerline. The inputs to the avionics are the locations of the three MLS ground units and the three observations of distance, azimuth angle, and elevation angle. When the three ground units are not collocated, this process requires an iterative algorithm. A March 1988 publication of the Radio Technical Commission for Aeronautics (RTCA) presents five such algorithms. These algorithms have various problems, such as errors in the analyses and/or computer code, and divergence or false solutions, etc., within the minimum coverage area of the MLS. This report, which identifies the various problems and offers corrections, concentrates on the two algorithms that assume a conical MLS azimuth antenna, as that is likely to be the only type that will be developed.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☐ UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT ☐ DTIC USERS	Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE (Include Area Code)	22c. OFFICE SYMBOL
Judith Schultz	(617) 271-8087	Mail Stop D135

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

8a.　　for Communications and Airspace Management Systems Program Office

Accession For	
NTIS CRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	

EXECUTIVE SUMMARY

    The Microwave Landing System (MLS) may be used for landing guidance,
for area navigation, and for computed-centerline approach. In the latter
functions, the MLS concept uses ground unit siting data, angle data from an
azimuth antenna and an elevation antenna, and range data from a distance
measuring equipment (DME). With these data, the avionics unit calculates
the aircraft's position in rectangular (Cartesian) coordinates relative to
the centerline of the selected runway. This process, called "position
reconstruction", is the subject of [1], which presents twelve algorithms
for various ground unit geometries and avionics computation options.

    Of these twelve algorithms, seven are restricted by the assumption
that the ground units are collocated in various combinations while five
assume general siting. The seven restricted algorithms are recognized in
[1] as not compliant with the minimum operational performance standards
(MOPS) and are not discussed further in this report, which examines the
five general (MOPS compliant) algorithms.

    Three of the five MOPS-compliant algorithms in [1] (Cases 7, 10, and
11) assume a planar-pattern azimuth antenna. It now appears that there may
not be any planar azimuth antennas, due to their significantly greater
complexity and cost. Two of these three (Cases 7 and 11) cannot be
generalized to handle the conical-pattern azimuth antenna case, and the
discussion of these two is accordingly abbreviated. The third, Case 10,
can be generalized to include the conical case; it then becomes essentially
identical to Case 9. This report therefore very briefly discusses these
three planar algorithms, 7, 10 and 11, but concentrates more on Cases 9 and
12, which are appropriate for conical-pattern azimuth antennas.

    Careful inspection and exercise [2] of the planar algorithms of Cases
7, 10, and 11 show that they have a variety of problems, including errors
in the mathematical analysis and/or the computer code. This report iden-
tifies the errors of text and code, and offers corrections.

    A similar inspection and exercise [2] of the conical cases 9 and 12,
show that they too have significant problems. Case 9, based on a Gauss-
Seidel approach, has minor errors in the computer code; a list of the
errata is provided. However, this case has more significant problems,
since its algorithm diverges at azimuth angles as small as 38° in some
geometries; this is within the minimum coverage of the system. Further,
the algorithm is very slow in many situations, some with azimuth angles
less than 30°. Correction of these problems of divergence and slow
convergence is difficult, and a satisfactory revision of the algorithm has
not been found.

Case 12 uses a Newton–Raphson approach. It, too, has problems of convergence. But whereas Case 9 exhibited difficulties at large azimuth angles, Case 12 has a region on and near the azimuth antenna boresight where false solutions and/or inappropriate error messages may occur; this is due to proximity to a singularity on the azimuth antenna boresight, and, therefore, on or near the runway centerline. The singularity has another effect; the dynamic range of a determinant which must be evaluated is of the order of 10^{11}. This dynamic range limits the choice of components that may be used in the avionics. In addition, this algorithm has the maximum number of possible false solutions. Finally, the algorithm is coded in a very general form, and requires 109 lines of FORTRAN and 13 internal DO-loops. A corrected computer code for Case 12 is offered; it corrects the major problems to the extent possible without changing the entire thrust of the approach. Several other Newton–Raphson algorithms are formed, all of which evade the difficulties noted above by using a different formulation of the underlying method. One of these is recommended as an alternate, and computer code for this algorithm is presented.

Appendixes present lists of errata, and computer code, for the several algorithms, the database with which the algorithms were tested, and examples of various initial condition algorithms for starting the iterative computation required in position reconstruction.

ACKNOWLEDGMENT

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ix

SECTION 1

INTRODUCTION

The Microwave Landing System (MLS) may be used both for landing guidance and for area navigation. In these functions, the MLS concept uses ground unit siting data, angle data from azimuth and elevation antennas, and range data from a distance measuring equipment (DME) to enable an avionics unit to provide the appropriate guidance information to the pilot or flight director. In area navigation, or in computed centerline operation, when the azimuth antenna is not on the runway centerline, the avionics must determine the aircraft's position in rectangular (Cartesian) coordinates relative to the centerline of the selected runway. This process, called position reconstruction, requires iteration when the three MLS ground units are not collocated. It is the subject of [1], which presents 12 position reconstruction algorithms.

These algorithms may be segregated into two classes, seven restricted, and five completely general. The seven restricted algorithms assume that the three ground units are collocated in various combinations; they are recognized in [1] as not compliant with the minimum operational performance standards (MOPS) for RNAV. They have not been studied and are not discussed further in this report. The five general algorithms assume that all three ground units may have separate and distinct sites and thus are MOPS-compliant. This report presents the findings of an examination of these five completely general algorithms in [1].

Three of the MOPS-compliant algorithms in [1] (Cases 7, 10, and 11) are specialized to the assumption of a planar azimuth antenna pattern. It now appears that there will probably be no planar azimuth antennas, due to their significantly greater complexity and cost. Two of the three (Cases 7 and 11) cannot be generalized to handle the conical azimuth antenna case, and the discussion of these two is accordingly abbreviated. The third, Case 10, can be generalized to include conical, but it then becomes essentially identical to Case 9. This report, therefore, very briefly discusses the three planar algorithms, 7, 10, and 11, but primarily concentrates on the two cases, 9 and 12, which are appropriate for conical pattern azimuth antennas.

Careful inspection and exercise of the three completely general planar algorithms, Cases 7, 10, and 11 in [1], show a variety of defects. Some of these defects appear to be typographical errors, while others are errors in the mathematical analysis and/or the computer code. This report identifies the errors of text and code, and offers corrections.

A similar examination and exercise of the two, more important, conical-pattern azimuth antenna algorithms, Cases 9 and 12, show that

1

too, have significant defects. Case 9, based on Gauss-Seidel techniques, has regions within the MLS coverage where it diverges, or converges slowly. It also has minor errors in the code. Corrections for the errors in the code are presented; however, the problems of slow convergence and divergence remain. Case 12, using a Newton-Raphson method, has a region on or near the azimuth antenna boresight where false solutions or unnecessary error messages may occur. This report notes where and why these algorithms have difficulties. Two new computer codes for Case 12 are offered: one corrects the major problems to the extent possible without changing the entire thrust of the approach, while an alternate avoids them by using a fundamentally different Newton-Raphson formulation.

Common elements, such as terminology and notation, are gathered in section 2. Section 3 presents a very brief discussion of the planar cases, (7), (10), and (11). Section 4 presents the examination of Case 9 of [1], while section 5 presents the examination of Case 12. Appendix A presents errata pages for all five algorithms, corrections for the code for Case 9, and both a simplified code and an alternate code for Case 12. Appendix B presents the details of the database which was used for exercising the algorithms. Appendix C presents several alternates for setting initial conditions.

SECTION 2

COMMON ELEMENTS

Common elements, such as notation and geometry, the mathematics, and an overview of the test database, are gathered in this section.

2.1 NOTATION, GEOMETRY, AND MATHEMATICS

The coordinate system for the problem is defined in figure 2.1. The x-axis is selected to be the runway centerline and its extension, with positive values towards the approach-end and negative values toward the stop-end of the runway. The origin of the coordinate system is arbitrarily selected, for the purposes of discussion, to be at the runway threshold, so that normal locations for the ground equipments have negative x-values. Thus, if the azimuth antenna is located near the stop-end of a 5000-foot runway, its x-value is approximately $x_A = -5000$, where the subscript A implies azimuth antenna. The positive direction of y lies to the left of an observer standing at the origin, with the stop-end of the runway behind him. The positive direction of z is up, completing the right-hand coordinate system.

The elevation angle is defined as positive counterclockwise, looking along the positive y-axis, so that positive angles correspond to positive altitude when the aircraft is within the coverage of the elevation antenna.

Azimuth is defined as positive clockwise from the x-axis, looking down towards the ground, contrary to the usual definition of a right-hand coordinate, but consistent with [1] and [3]. It is assumed that the azimuth antenna electronic boresight is parallel to the runway centerline. This does not imply any loss of generality, for a simple rotation about the z-axis enables a general orientation of the boresight.

The notation is defined as follows:

x, y, z	Components of estimated position of aircraft
x_A, y_A, z_A	Components of location of azimuth antenna
x_D, y_D, z_D	Components of position of DME
x_E, y_E, z_E	Components of position of elevation antenna
x_T, y_T, z_T	Components of true position of the aircraft
x_0, y_0, z_0	Components of the initial position estimate used in the iterative procedures

θ	Azimuth angle relative to the runway centerline, measured at the aircraft in radians (unless otherwise stated)
ϕ	Elevation angle relative to the horizon, measured at the aircraft in radians (unless otherwise stated)
ρ	Slant range from the DME transmitter, measured at the aircraft in feet.

Figure 2-1 shows the geometry of a completely general arrangement.
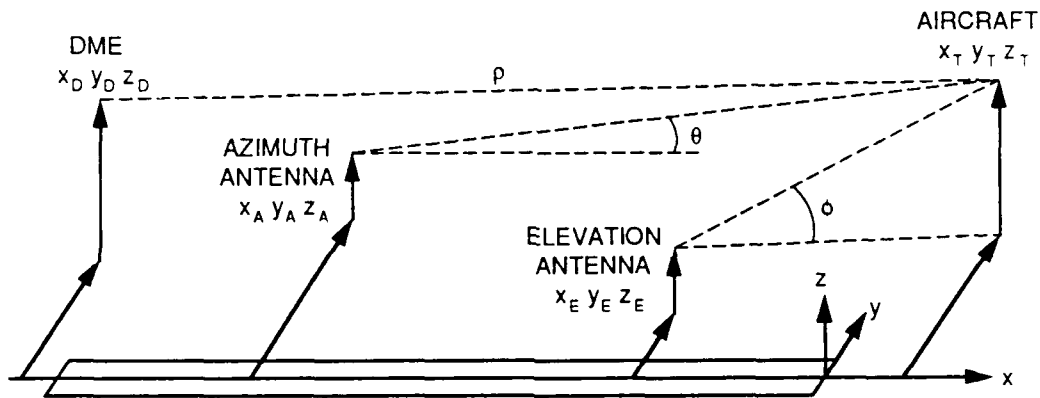
Figure 2-1. Geometry

The mathematics of the problem are now presented. The distance from the DME is

$$\rho = [(x_T-x_D)^2 + (y_T-y_D)^2 + (z_T-z_D)^2]^{1/2}. \qquad (2\text{-}1)$$

The azimuth angle is

$$\tan\theta = -(y_T-y_A)/[(x_T-x_A)^2 + (1-I56)(z_T-z_A)^2]^{1/2}. \qquad (2\text{-}2)$$

where $I56 = 0$ for conical antennas, and 1 for planar antennas, see [3].

The elevation angle is

$$\tan\phi = (z_T-z_E)/[(x_T-x_E)^2 + (y_T-y_E)^2]^{1/2}. \qquad (2\text{-}3)$$

4

In (2-1) through (2-3), replace x_T, y_T and z_T by x, y, and z, respectively. The mathematical problem of position reconstruction is to solve for the unknowns, x, y, and z, given the data and the observations.

2.2 TEST DATABASE

This subsection discusses briefly the sample set of ground station siting arrangements and the sample set of true aircraft positions selected for exercising the algorithms.

A set of five ground station siting arrangements was considered. These include conventional and unconventional arrangements. All three ground units are collocated at the usual elevation antenna placement in site 1, as in a conventional heliport. Site 2 is a conventional split-site, with the DME at the azimuth antenna, while site 3 is also split, but the DME is collocated with the elevation antenna, using a suggestion of [2]. Sites 4 and 5 are fully dispersed, to provide a general basis for testing the algorithms. While sites 1 and 2 are conventional, the other three cover a wider range of possibilities.

A set of 75 aircraft true locations was used in the study. These locations were selected to test the extremes of range and angles, and to cover both normal and unusual situations so as to test all realistic normal and special cases. The behavior of each algorithm was tested at each aircraft true location for all five ground unit positions.

The details of the database are provided in appendix B.

Figure 2.2 presents the five ground unit site configurations; figures 2.3 and 2.4 show the plan views of the 75 aircraft true locations used for exercising Cases 9 and 12, respectively. The numerals enable identifying the locations exactly, by referring to appendix B. Aircraft locations 1 through 50 are common to the examination of cases 9 and 12. For Case 9, the regions of degraded performance are at large angles, while for Case 12, the region of degraded performance is at very small azimuth angles. Therefore aircraft locations 51 through 75 are concentrated at larger azimuth angles for Case 9, and at and near an azimuth of zero for Case 12.
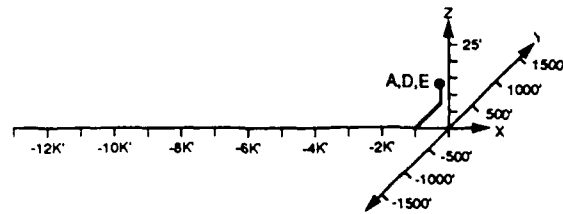
GROUND SITE CONFIGURATION #1
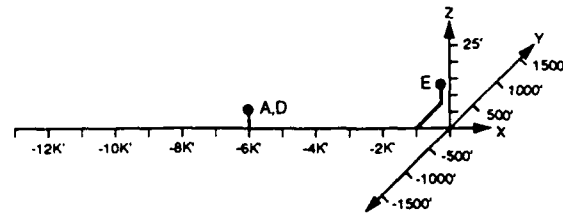
GROUND SITE CONFIGURATION #2
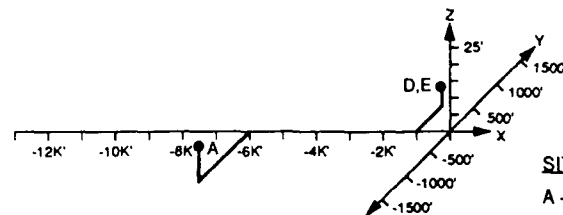
SITES

A – AZIMUTH ANTENNA

D – DME

E – ELEVATION ANTENNA

GROUND SITE CONFIGURATION #3
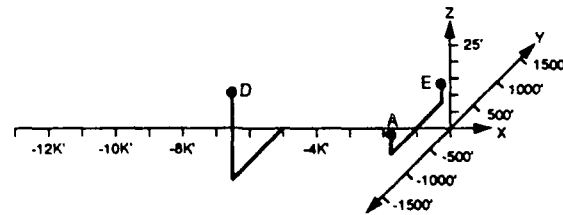
GROUND SITE CONFIGURATION #4
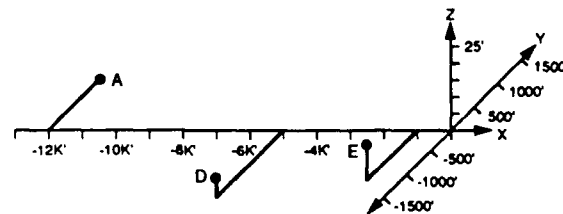
GROUND SITE CONFIGURATION #5

Figure 2-2.   Ground Unit Configuration Data

6
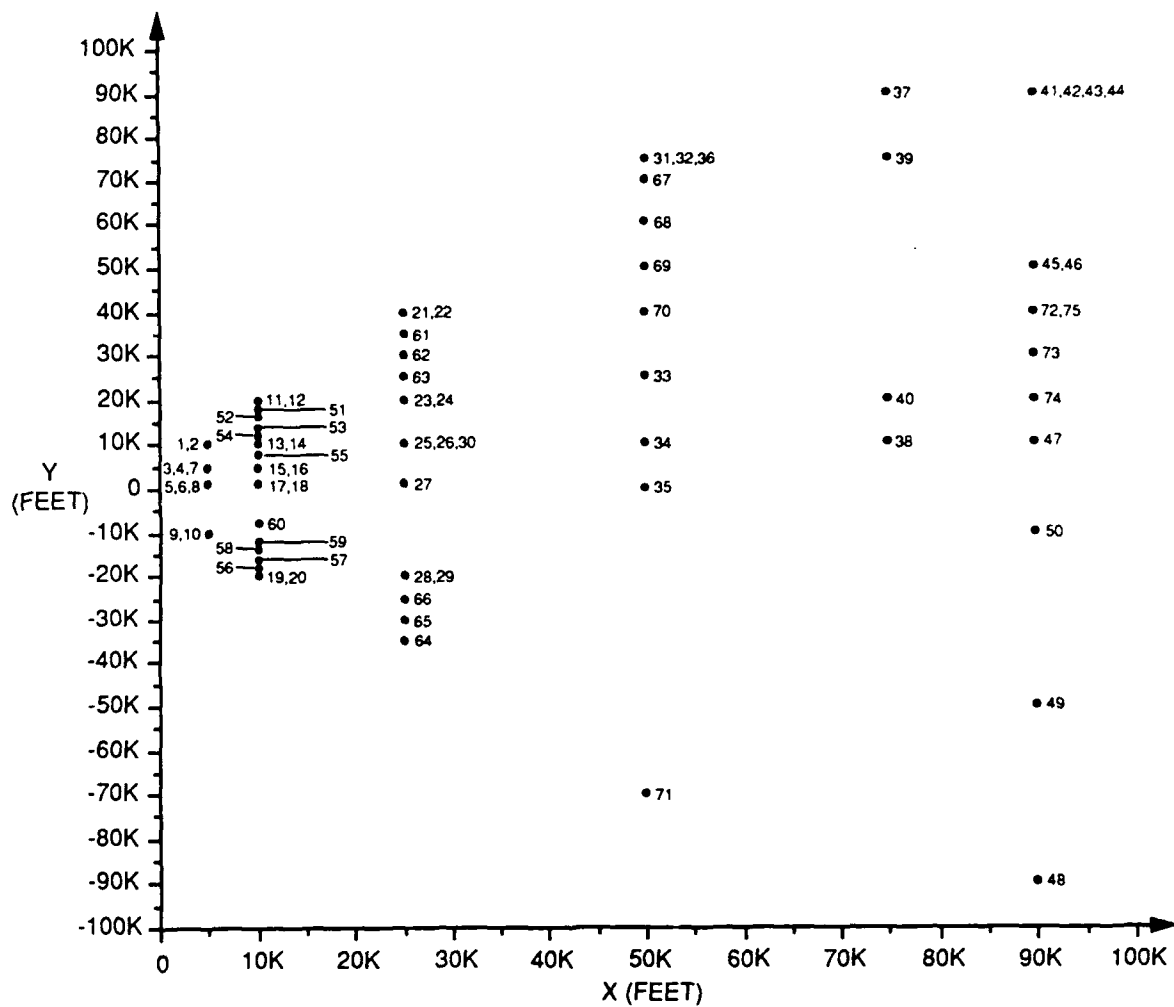
Figure 2-3. Plan View of Aircraft Locations for Case 9

7
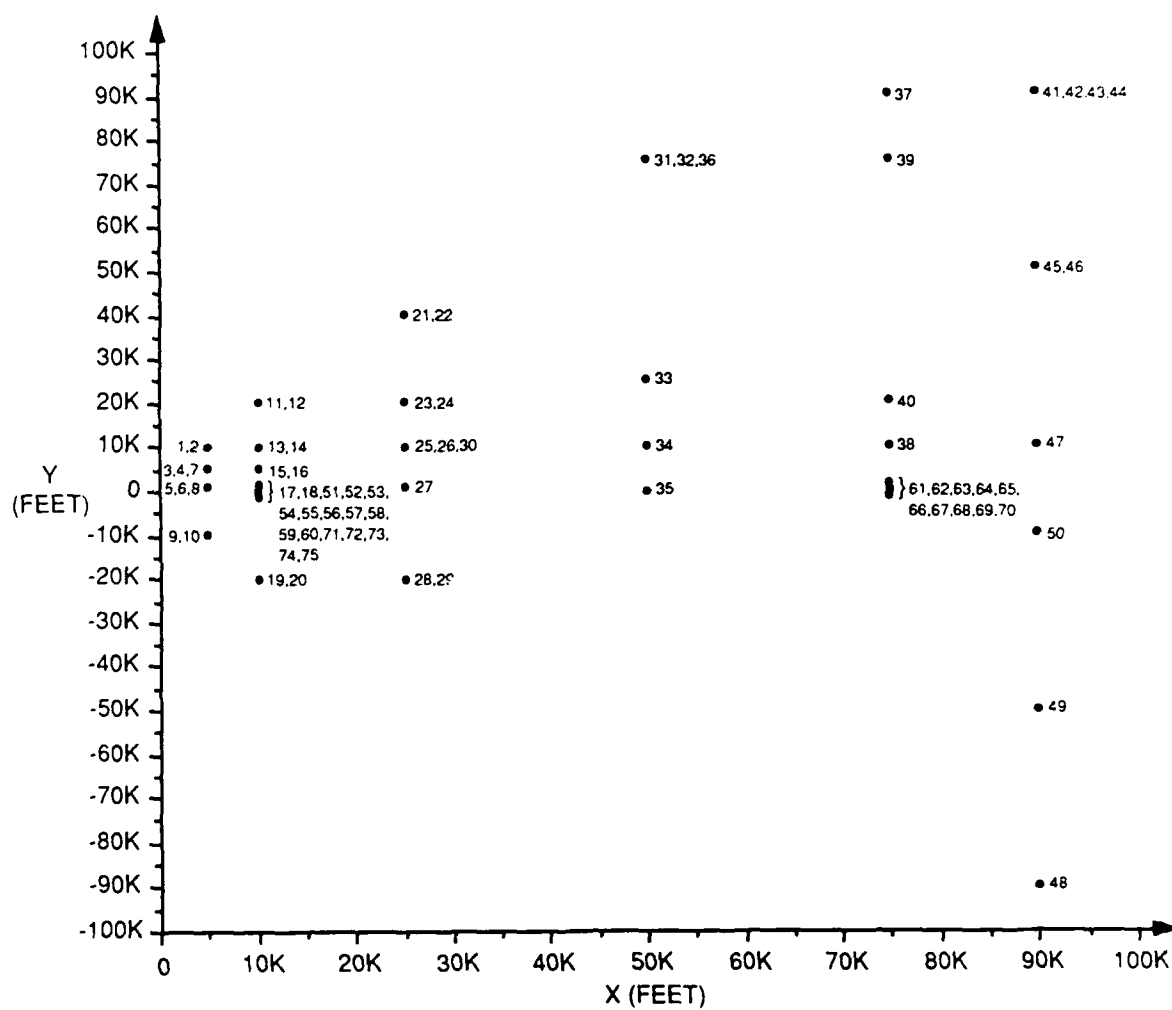
Figure 2-4. Plan View of Aircraft Locations for Case 12

8

SECTION 3

PLANAR AZIMUTH ANTENNA ALGORITHMS

There are three planar algorithms in [1]. This section shows why two of them (Cases 7 and 11) cannot be generalized to conical, and how the third (Case 10) can be generalized.

3.1 CASES 7 AND 11

These two cases in [1] cannot be generalized to handle the conical azimuth antenna situation. From (2-2), the planar azimuth antenna is characterized by the linear relationship

$$y = y_A - (x - x_A)\tan\theta_P \qquad (3-1)$$

obtained from (2-2) by setting $I56 = 1$ and $\theta = \theta_P$, where θ_P is the planar angle observation. However, the conical azimuth antenna is characterized by the nonlinear relationship

$$y = y_A - [(x - x_A)^2 + (z - z_A)^2]^{1/2}\tan\theta_C. \qquad (3-2)$$

obtained from (2-2) by setting $I56 = 0$ and $\theta = \theta_C$. As (3-1) is linear, it may be inverted to yield the linear relationship

$$x = x_A - (y - y_A)\cot\theta_P. \qquad (3-3)$$

Case 7 uses (3-3) in the range expression (2-1), which then yields a quadratic equation in y as a function of z. This quadratic is solved for y, and then (3-3) is solved for x with this value of y, and (2-3) is solved for z using these new values of x and y. Case 11 takes the opposite choice, using (3-1) in (2-1) to eliminate y, etc. But comparison of (3-1) or (3-3) with (3-2) shows that the simplification offered by the former two is not available with (3-2), thus preventing generalization.

Errata lists developed from the analyses of Cases 7 and 11 are presented in appendix A.

3.2 CASE 10

This case can be generalized to include the conical azimuth antenna. The planar azimuth antenna equation (3-1) may be formed from the conical equation (3-2) simply by omitting the term $(z-z_A)^2$ from (3-2). This process may be reversed, and the planar case can therefore be generalized. Combine (3-1), the planar equation for lateral position, and (3-2), the conical equation, using the data bit $I56$, from [1] or [3], as

$$y = y_A - [(x - x_A)^2 + (1 - I56)(z - z_A)^2]^{1/2}\tan\theta \qquad (3-4)$$

where I56 is 0 for conical, and 1 for planar antennas. This change generalizes case 10, which then becomes almost identical to Case 9, below, and thus requires no further discussion.

An errata list for Case 10 is presented in appendix A.

SECTION 4

CASE 9: A GAUSS-SEIDEL CONICAL-AZIMUTH ALGORITHM

Case 9 of [1] is a conventional Gauss-Seidel algorithm for conical azimuth antennas. A clear exposition of the theory of Gauss-Seidel iteration is presented in [5], including discussion of the conditions for divergence and for slow convergence. The principles of Gauss-Seidel iteration for nonlinear equations are outlined heuristically in the MLS context. Initial values are assumed for two of the three variables (for example, x_0 and y_0). With these values, the equation for the third variable, z, is evaluated. This new value, z_1, and one of the initial conditions (say, x_0), are used to evaluate the equation for the second variable, y. And with these two new values, z_1 and y_1, the last equation is evaluated for the third variable, x, yielding x_1. The process is then iterated until the solution is of acceptable accuracy. In some situations, Gauss-Seidel iteration can be unstable or divergent.

The mathematical structure for Case 9 is defined as

Initial conditions for x_0 and y_0 (z_0 is not needed) are

$$x_0 = \rho\cos\theta, \quad y_0 = -\rho\sin\theta. \tag{4-1}$$

Iterative solution for altitude; solve (2-3) to find

$$z_{i+1} = z_E + [(x_i-x_E)^2 + (y_i-y_E)^2]^{1/2}\tan\phi. \tag{4-2}$$

Iterative solution for lateral position; solve (2-2) to find

$$y_{i+1} = y_A - [(x_i-x_A)^2 + (1-I56)(z_{i+1}-z_A)^2]^{1/2}\tan\theta. \tag{4-3}$$

Iterative solution for along-runway position; solve (2-1) for x as

$$x_{i+1} = x_D + [\rho^2 - (y_{i+1}-y_D)^2 - (z_{i+1}-z_D)^2]^{1/2}. \tag{4-4}$$

Following [1], the positive value of the radical in (4-4) is required in the present context, where $x \geq x_D$. Further, it is essential that the radicand in (4-4) be calculated separately before taking the square root, since the radicand is negative for values of y and z which can be reached during iteration.

The sequence of equations follows the usual Gauss-Seidel procedure, using the equations in the order of increasing magnitudes of the usual values of the slopes, and using each newly-estimated value as soon as it is available; this normally leads to rapid convergence.

11

When the azimuth angle is relatively small, this algorithm converges well; however, it requires more than five iterations to converge close enough to the correct solution in many cases with azimuth angles between 22° and 30°. Further, the initial condition used above and in [1] sometimes causes unnecessary entry into the negative-radicand situation mentioned above. Moreover, the algorithm diverges in many geometries, one with azimuth angle as low as 38°. This case, 3-9 (ground unit arrangement 3, aircraft location 9), has the elevation antenna and DME collocated at the normal site for the elevation antenna, a possibility suggested in [2]. In addition, this case exhibits an immediate negative-radicand problem both with the given initial condition and with an alternate ($x_0 = y_0 = z_0$). Table 4-1 shows the behavior of this algorithm in the case 3-9, cited above. Because of the immediate negative-radicand problem mentioned above, the initial condition, in the second column, has been selected very close to the true location in the first column to display clearly the divergent behavior in the next six columns.

Table 4-1. Divergence in Case 3-9

GROUND STATION SITE GEOMETRY # 3

AZIMUTH ANTENNA SITE			DME TRANSMITTER SITE			ELEVATION ANTENNA SITE		
X	Y	Z	X	Y	Z	X	Y	Z
-6000.	-1000.	10.	-1000.	500.	5.	-1000.	500.	5.

AIRCRAFT POSITION # 9. OBSERVED DATA: RHO = 12588.3 THETA = 37.95 PHI = 16.12

ITERATION NUMBER I	TRUE POS.	INIT. POS.	1	2	3	4	5	6
X	5000.00	5500.00	4061.68	6092.74	3264.82	6824.62	1337.77	8210.56
Y	-10000.00	-10500.00	-10418.31	-9301.11	-10815.98	-8720.97	-11365.27	-7336.81
Z	3500.00	.00	3697.55	3482.99	3501.42	3499.88	3500.01	3500.00

Slow convergence is discussed further. An allowance of 0.017° for Path Following Noise (PFN) is given in [3] for the avionics' receiver. The receiver performance is limited by physical considerations such as noise and power, whereas the algorithm is, in principle, capable of almost-perfect performance. An allowance of 0.017°/3 for error of the algorithm, approximately one part in 10,000, is negligible by comparison, when combined RSS. It was therefore assumed that convergence is slow if the magnitude of the error in any variable exceeds ($x_T/10,000$) after five iterations. Alternate criteria, such as Control Motion Noise (CMN), lead to even smaller allowable error allowances. The regions of convergence, slow convergence, and divergence are shown in figures 4-1 through 4-5, for ground site configurations 1 through 5, respectively. In each figure, the ground site configuration is presented as an inset to ease interpretation.
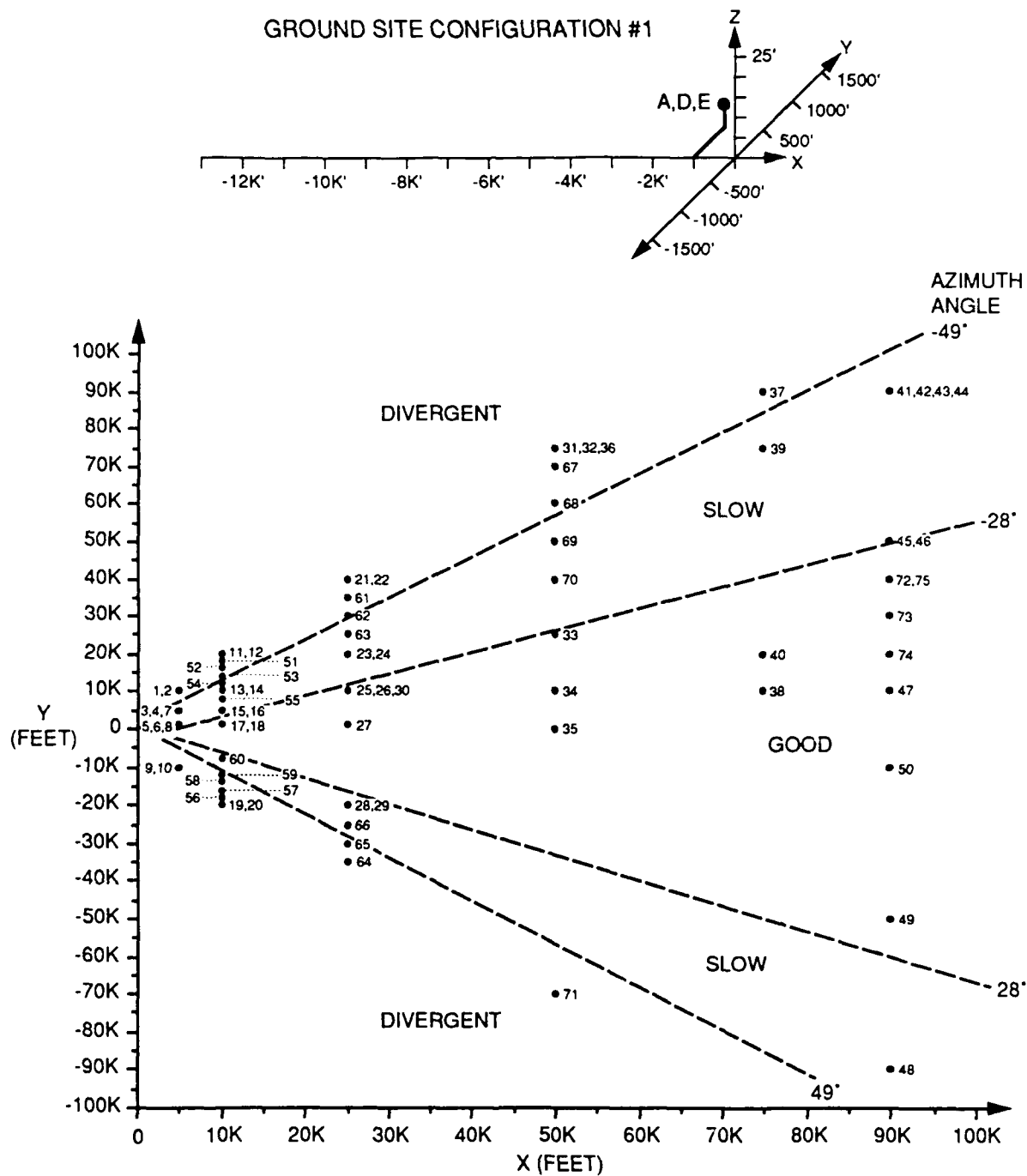
Figure 4-1. Performance Regions for Case 9, Ground Unit Siting 1

13
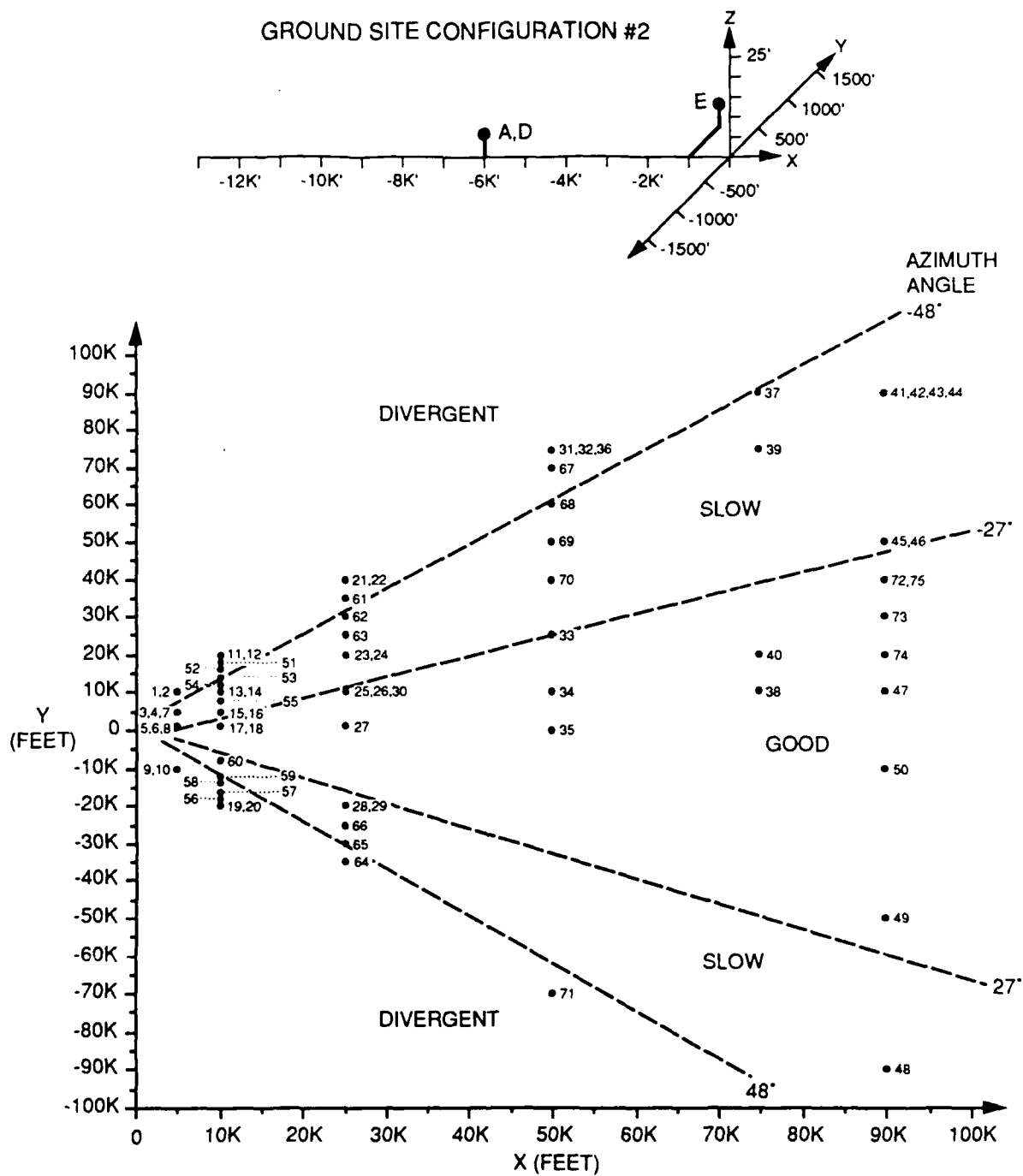
Figure 4-2. Performance Regions for Case 9, Ground Unit Siting 2
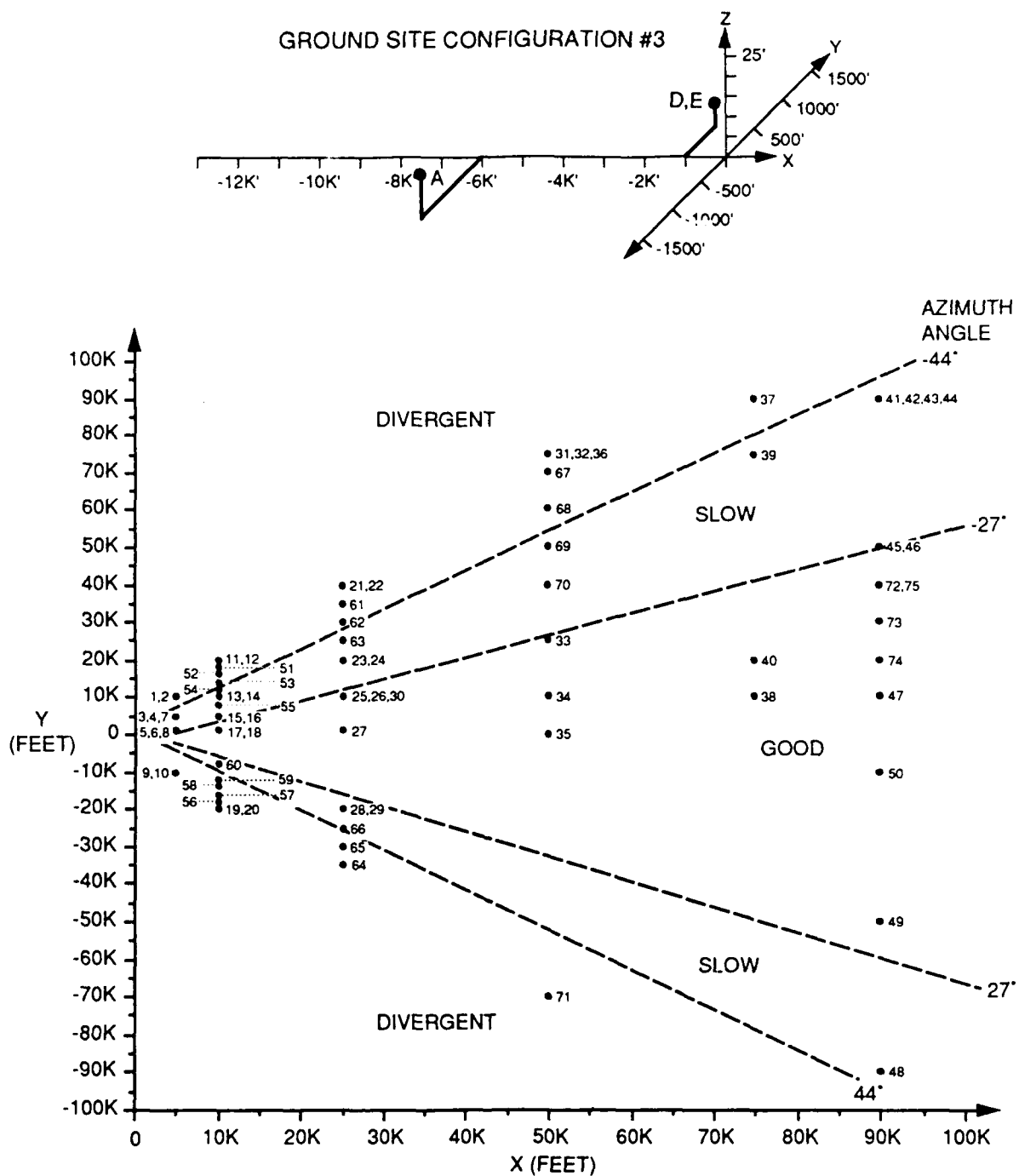
Figure 4-3.   Performance Regions for Case 9, Ground Unit Siting 3
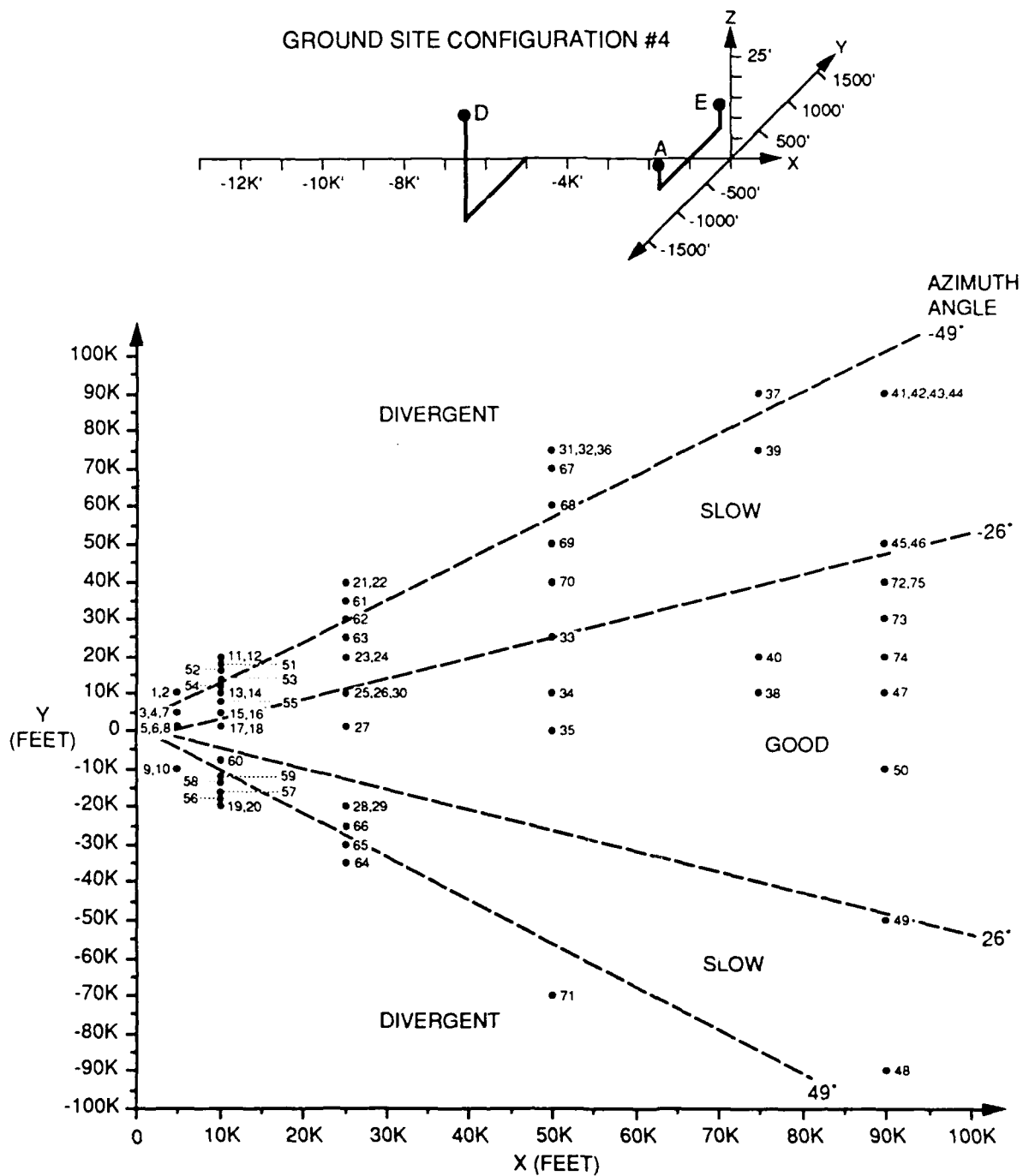
Figure 4-4. Performance Regions for Case 9, Ground Unit Siting 4
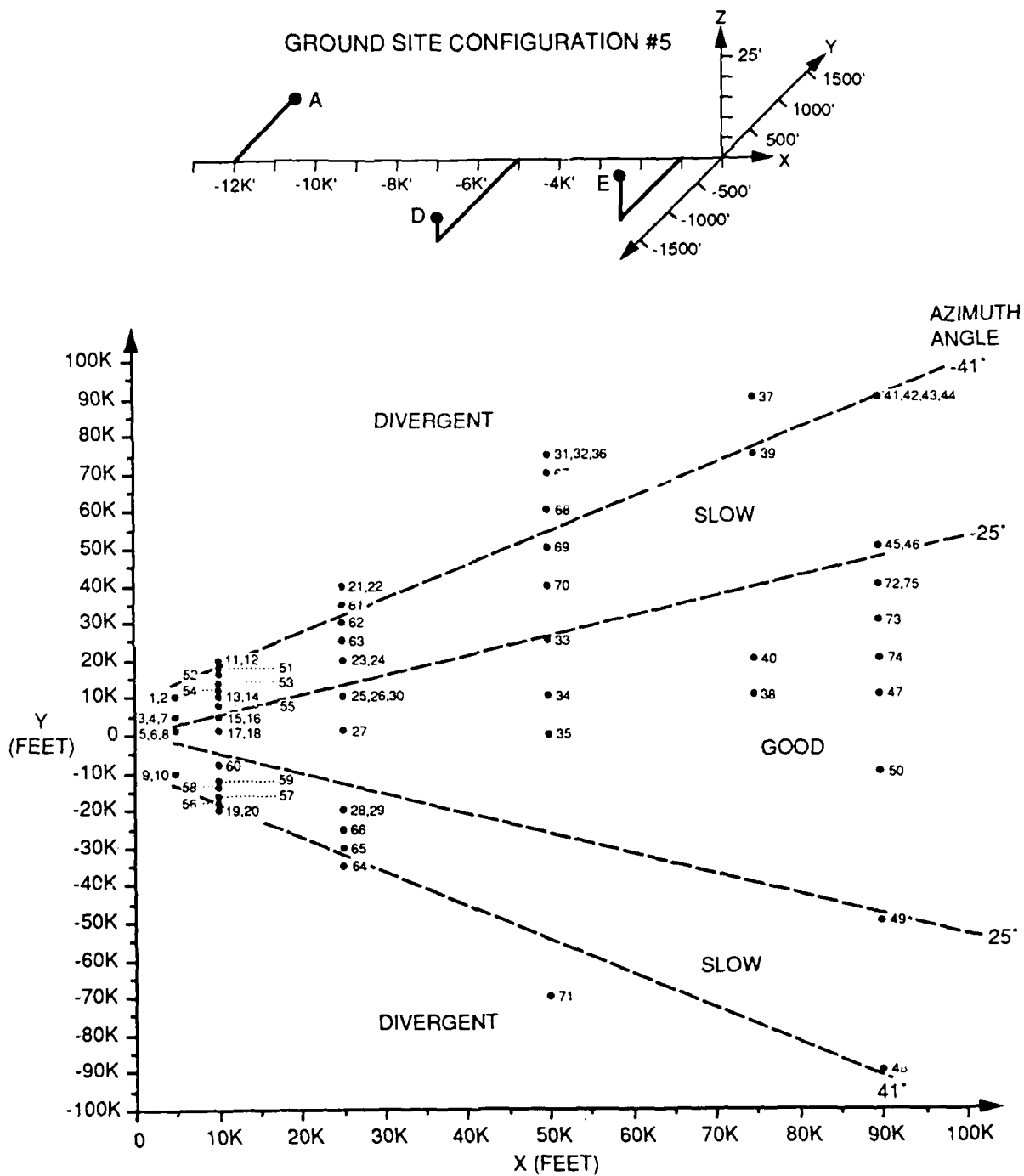
Figure 4-5. Performance Regions for Case 9, Ground Unit Siting 5

In figures 4-1 through 4-5, the boundary lines between the good, slow, and divergent regions should not be viewed as exact, but as somewhat fuzzy, since stability and speed of convergence are influenced by altitude, which does not show on these figures. They are also influenced by ground unit site geometry, although that becomes less influential at longer distances from the threshold. Thus, for example, in figure 4-3, location 9 (mentioned above), is divergent with an azimuth angle of 37.95°, while location 59, with azimuth angle of 33.28°, is slow. The observed azimuth angles between the several regions are marked approximately on these figures. These are very rough boundaries; for example, Case 3-9, cited above, is unstable and is thus outside the marked boundary for divergence, but its azimuth angle is less than 38°, due to the altitude-effect on the azimuth angle observation, while the diagram indicates that the azimuth angle of its horizontal-plane projection exceeds 44°.

These figures show the characteristic behavior of this algorithm. The existence of regions of slow convergence or of divergence are not the consequence of errors of analysis or code, but show the inherent nature of this algorithm and of this form of Gauss-Seidel iteration. If these properties are unacceptable, an entirely different type of algorithm is required. Diagrams similar to figures 4-1 through 4-5 appear in section 5 where the Newton-Raphson method, Case 12, is discussed. Comparison of figures 4-1 through 4-5 with 5-1 through 5-5 shows the characteristic difference between these two algorithms. At large azimuth angles, where Case 9 is slow or divergent, Case 12 converges successfully. On the other hand, at very small azimuth angles, where Case 9 is convergent, Case 12 has problems; the nature of these problems and their cause are discussed in section 5.

A list of errata is presented in appendix A, that, to the extent possible, corrects the defects of the algorithm as presented in [1] and enabled developing the results shown in this report. However, as remarked above, these corrections only enable the algorithm to run correctly, and do not correct the basic reasons for its slow convergence and divergence at large azimuth angles.

SECTION 5

CASE 12: A NEWTON-RAPHSON CONICAL-AZIMUTH ALGORITHM

Case 12 uses the well-known and usually reliable Newton-Raphson (NR) method of iterative solution. This method is briefly described in the MLS context. Assume three functions, f, g, and h, in the three variables of x, y, and z. The solution requires finding one or more sets of values of x, y, and z, such that $f = g = h = 0$ at the solution. Given three initial conditions, the increments added at each iteration to x, y, and z are found by a matrix procedure as

$$\begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix}_{i+1} = -(J^{-1}) \begin{pmatrix} f \\ g \\ h \end{pmatrix}_i \qquad (5\text{-}1)$$

where J is the (Jacobian) matrix of the partial derivatives of f, g, and h with respect to the variables x, y, and z. The right-hand side of (5-1) is evaluated at iteration i. The process fails if the matrix J is singular with determinant equal to zero, or when the matrix is nearly singular and the determinant is so small that round-off errors become important. And, it may fail to converge, or converge to a wrong solution if the desired solution is separated by a maximum or a minimum from the (x_T, y_T, z_T) triple at any point in the iteration. This concept is applied to the MLS problem.

5.1 PROPERTIES OF CASE 12

Return to (2-1)-(2-3); the approach of [1] squares these equations to eliminate the radicals, and then multiplies (2-2) and (2-3) by $\cos^2\theta$ and $\cos^2\phi$, respectively, to form f, g, and h as

$$f = (x-x_D)^2 + (y-y_D)^2 + (z-z_D)^2 - \rho^2 \qquad (5\text{-}2)$$
$$g = -(x-x_A)^2\sin^2\theta + (y-y_A)^2\cos^2\theta - (z-z_A)^2\sin^2\theta \qquad (5\text{-}3)$$
$$h = -(x-x_E)^2\sin^2\phi - (y-y_E)^2\sin^2\phi + (z-z_E)^2\cos^2\phi. \qquad (5\text{-}4)$$

The Jacobian matrix of partial derivatives, J, is

$$J = \begin{pmatrix} 2(x-x_D) & 2(y-y_D) & 2(z-z_D) \\ -2(x-x_A)\sin^2\theta & 2(y-y_A)\cos^2\theta & -2(z-z_A)\sin^2\theta \\ -2(x-x_E)\sin^2\phi & -2(y-y_E)\sin^2\phi & 2(z-z_E)\cos^2\phi \end{pmatrix}. \qquad (5\text{-}5)$$

This algorithm may be generalized to include the case of a planar azimuth antenna by multiplying by (1-156) the $[(z-z_A)^2\sin^2\theta]$ term of g in (5-3) and the (2,3) term $[-2(z-z_A)\sin^2\theta]$ in (5-5).

The problems of this implementation are now discussed.

19

5.2 PROBLEMS WITH CASE 12

It was remarked above that the problem cannot be solved when the matrix J is singular; this is recognized in [1], which uses an error message whenever the matrix is singular. But in the important special case when the aircraft is within the avionics' quantizing region for azimuth of zero, the matrix (5-5) is singular, for all elements on the second row vanish at the solution, when $y = y_A$. It is not desirable that the avionics produce an error message when the aircraft is on the azimuth antenna boresight, especially since an exact solution is available.

Another aspect of this problem of singularity is that when the azimuth angle is not zero, but is merely very small, it is possible for round-off errors or similar, normally negligible, computational anomalies to cause the solution to converge to a false location. If the ground units are collocated, the singularity condition will occur if the estimated position of the aircraft is on the azimuth antenna boresight at any stage of the iteration process. A complete fix for this singularity problem, within the context of the approach of [1], is not evident.

Further, the matrix can be singular if the initial conditions are not carefully selected. For example, assume the three units are so oriented that $y_D = y_A = y_E$; then the initial condition set may not use $y_0 = y_D$, for that would cause the second column of J to be zero, resulting in an unnecessary error message. This problem may be viewed as contained within the problem of the preceding paragraphs, concerning the singularity case. Further, if $z_D = z_A = z_E$, then $z = z_E$ may not be used as an initial condition, for that would cause the third column of (5-5) to equal zero, resulting again in a singularity problem.

In (5-2)-(5-4), the sines and cosines of both angles appear as squares. Therefore, since $\sin^2(-\theta) = \sin^2\theta$, $\cos^2(\theta+180) = \cos^2\theta$, etc., the number of possible solutions becomes greater (eight solutions have been shown), so that convergence to the correct solution depends on having a good initial condition, and on the solution history at each iteration remaining sufficiently distant from all singularities. Satisfactory initial conditions to avoid this situation are offered in [1], and are also given in appendix C.

The determinant of the square matrix in (5-5) has a very large dynamic range. Assume collocation of the three ground units at the origin; then the determinant has the value $\Delta = 8xyz$. The singularities at $x = x_D$ and at $z = z_E$ are out of coverage for Category II operations, and are thus not of immediate interest to the USAF at present. However, the singularity on the antenna boresight, at $y = y_A$, is within coverage, as mentioned. Assume

20

the following limits, consistent with Category II, on the variables:

a. $x_{MAX} = 10^5$ feet

b. $x_{MIN} = 10^3$ feet

c. $y_{MAX} = 10^5*\tan\theta_{MAX}$ where $\theta_{MAX} = 60°$

d. $y_{MIN} = 10^3*\tan\theta_{MIN}$ where $\theta_{MIN} = 0.005°$, half the avionics' resolution requirement of [4], and treating any case where $|y|$ subtends any angle of less than $0.005°$ as within the singularity special case mentioned above

e. $z_{MAX} = 2*10^4$, and $\phi_{MAX} = 30°$, from [3]

f. $z_{MIN} = 10^3*\tan(1°)$, as the minimum coverage in elevation is $0.9°$.

Then the dynamic range of the determinant ($\Delta_{MAX}/\Delta_{MIN}$) is approximately $2.3*10^{11}$. This dynamic range will have an impact on the avionics because of the limited variety of computer chips at present available to carry out the required matrix inversion operation.

Finally, the code is given in a very general format which uses 109 lines of code and 13 DO-loops.

Two approaches to improving this algorithm are presented:

a. The code is rewritten to treat directly the problem of singularity in a more compact and simple form, to the extent possible without fundamentally changing the approach. This is called the simplified form.

b. The Newton-Raphson method is implemented, below, in a way that avoids completely the problems cited above, in what is called the alternate form.

5.3 SIMPLIFIED NEWTON-RAPHSON APPROACH

This approach simplifies the code and eliminates the singularity problem when the observed quantized azimuth angle is exactly zero. This is accomplished by using a logic expression to recognize when the observed azimuth angle is zero; then set $y = y_A$ and $g = 0$, omit the second row and second column of (5-5), omit g, and solve the reduced problem as a function of x and z. Code for this revised form of the algorithm is shown in appendix A. However, this does not eliminate the problems when the solution is near the singularity. Table 5.1 shows a case where the

21

algorithm converges to a wrong solution due to proximity to the
singularity. The simulation was single-precision; it is not desirable to
require that the avionics be double-precision unless it is truly essential.
A list of the errata appears in appendix A, together with code to correct,
or ameliorate, some of the defects noted above.

Table 5-1. Failure to Converge with Case 12
Simplified, Due to Proximity to the Singularity


```
                         GROUND STATION SITE GEOMETRY # 3
      AZIMUTH ANTENNA SITE      DME TRANSMITTER SITE          ELEVATION ANTENNA SITE
       X       Y      Z           X       Y      Z              X       Y       Z
     -6000.  -1000.  10.        -1000.   500.    5.          -1000.   500.    5.


   AIRCRAFT POSITION # 70.   OBSERVED DATA: RHO = 76061.1    THETA = .18   PHI = 1.88
        TRUE POS.    INIT. POS.
   ITERATION NUMBER I       1         2         3         4         5         6
   X  75000.00  76060.72  75020.41  75010.57  75009.91  75009.87  75009.87  75009.87
   Y  -1250.00   -234.64   -576.48   -714.43   -747.76   -749.96   -749.97   -749.97
   Z   2500.00   2495.00   2500.01   2500.00   2500.00   2500.00   2500.00   2500.00
```


Figures 5-1 through 5-5 show the behavior of the simplified form of
Case 12. Problems, such as in case 3-70 noted above, occur with this
Newton-Raphson approach only on or very near the boresight of the azimuth
antenna. The problem cases appear in the figures as open squares, and the
pertinent specific case-number is underlined, to enable identification of
the specific case by use of the database in appendix B. These figures may
be compared with those for the Gauss-Seidel approach of Case 9 in figures
4.1 through 4.5, as discussed previously.

5.4 ALTERNATE NEWTON-RAPHSON APPROACH

The problems observed, above, in the original and simplified versions
of this algorithm are the singularity on the azimuth antenna centerline,
the presence of multiple solutions, and perhaps the dynamic range of the
determinant. The common source of these problems is the presence of the
squared terms in (5-3) and (5-4). Any useful alternate must eliminate the
two major problems, and therefore must not use the squared forms used as in
(5-3) and (5-4). Some alternate possibilities are outlined below; all
these alternates have two solutions of which only one is normally within
coverage, and none is known to have any singularity within coverage.
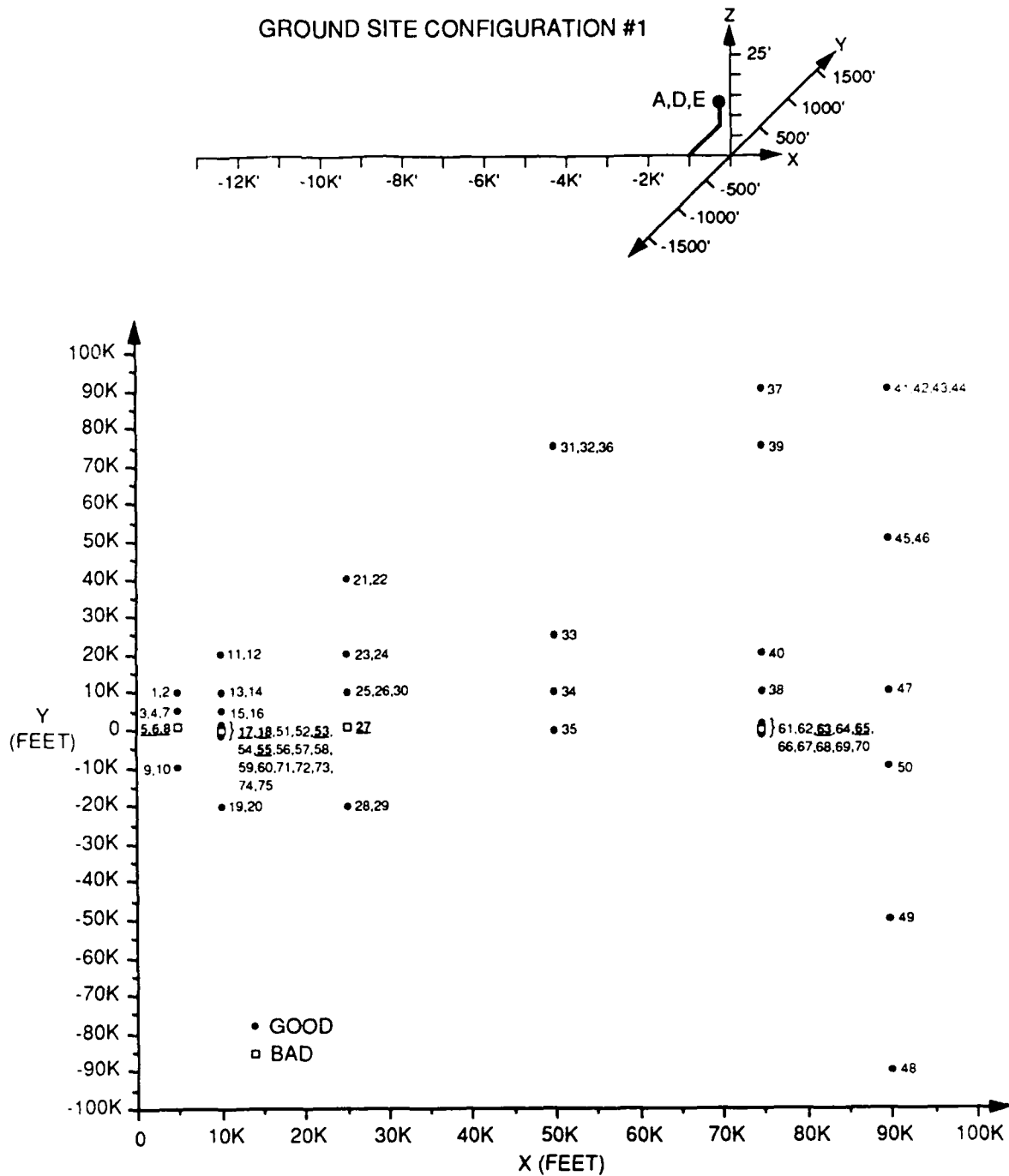
Figure 5-1. Performance Regions for Case 12 Simplified,
Ground Unit Siting 1
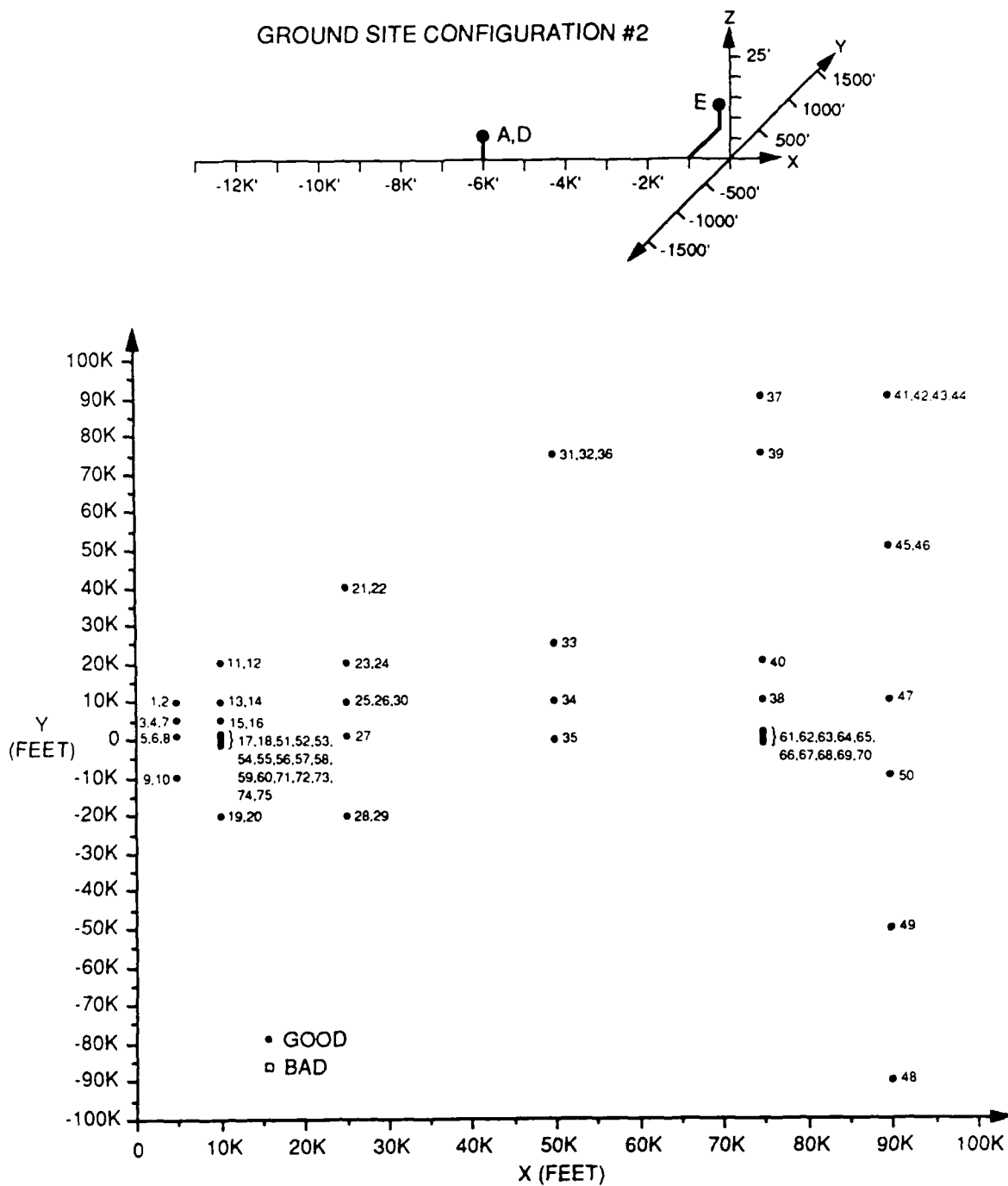
Figure 5-2. Performance Regions for Case 12 Simplified,
Ground Unit Siting 2

GROUND SITE CONFIGURATION #3

Z

25'

Y

1500'

D,E ●

1000'

500'

X

-12K'   -10K'   -8K ● A -6K'   -4K'   -2K'   -500'

-1000'

-1500'

Y (FEET)

100K

90K   ● 37   ● 41,42,43,44

80K   ● 31,32,36   ● 39

70K

60K

50K   ● 45,46

40K   ● 21,22

30K   ● 33

20K   ● 11,12   ● 23,24   ● 40

10K   1,2 ●   ● 13,14   ● 25,26,30   ● 34   ● 38   ● 47

3,4,7 ●   ● 15,16

0   5,6,8 ● □ } 17,18,51,52,53,   ● 27   ● 35   □ } 61,62,63,64,65,
              54,55,56,57,58,   66,67,68,69,70

-10K 9,10 ●   59,60,71,72,73,   ● 50
              74,75

-20K   ● 19,20   ● 28,29

-30K

-40K

-50K   ● 49

-60K

-70K

-80K   ● GOOD

-90K   □ BAD   ● 48

-100K
     0   10K   20K   30K   40K   50K   60K   70K   80K   90K   100K

X (FEET)
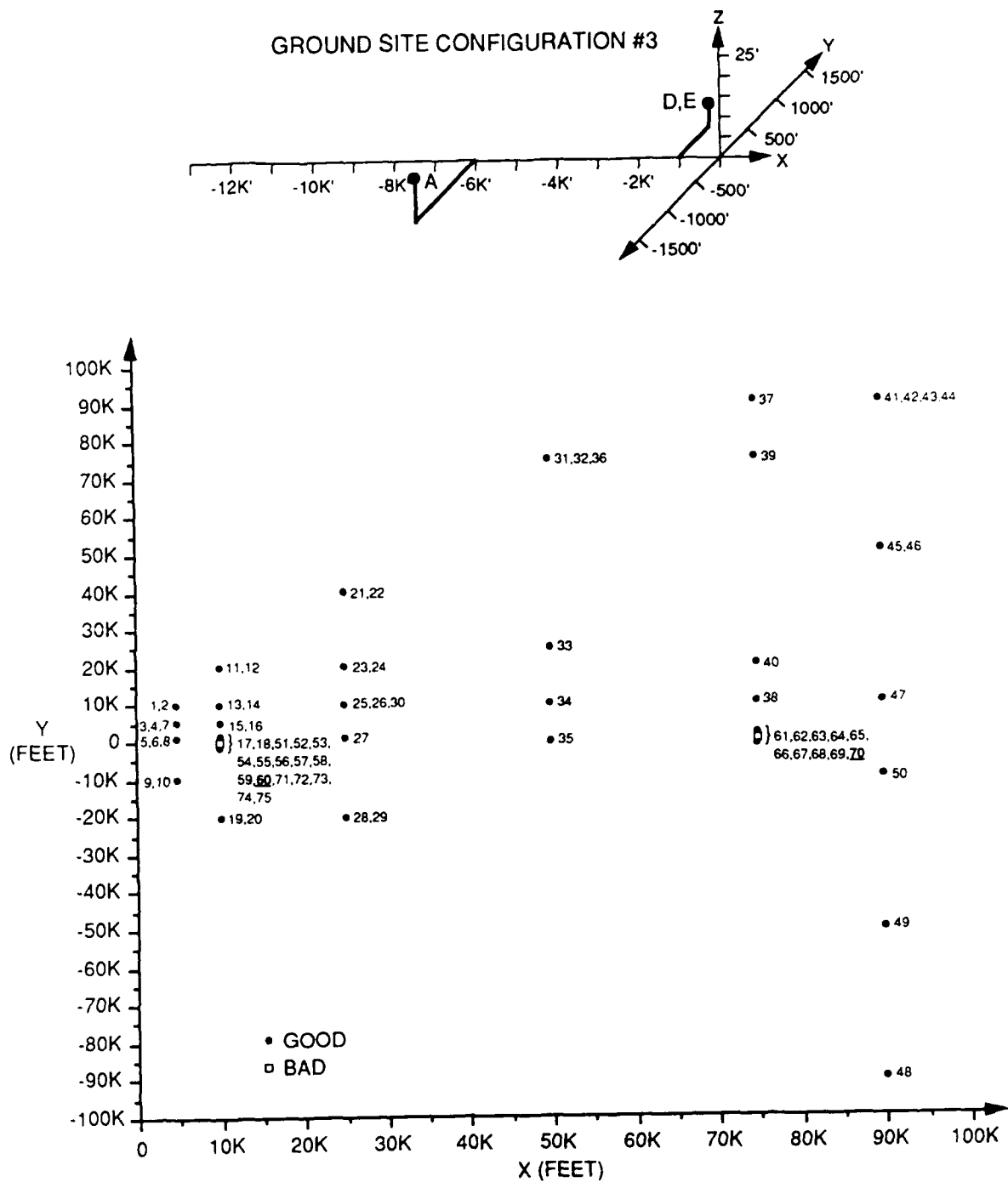
Figure 5-3.   Performance Regions for Case 12 Simplified,
Ground Unit Siting 3

GROUND SITE CONFIGURATION #4
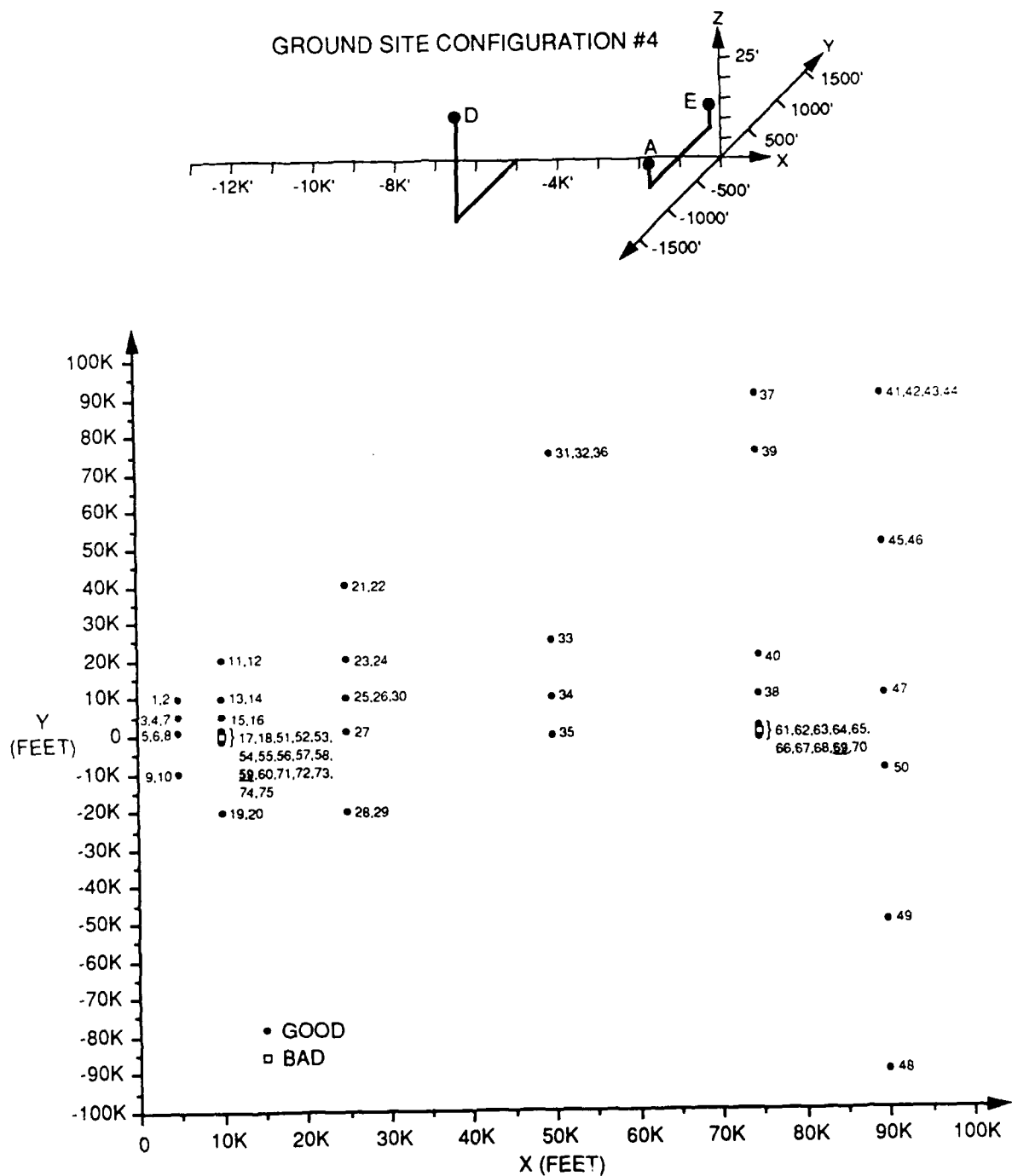
Figure 5-4.   Performance Regions for Case 12 Simplified,
Ground Unit Siting 4

GROUND SITE CONFIGURATION #5
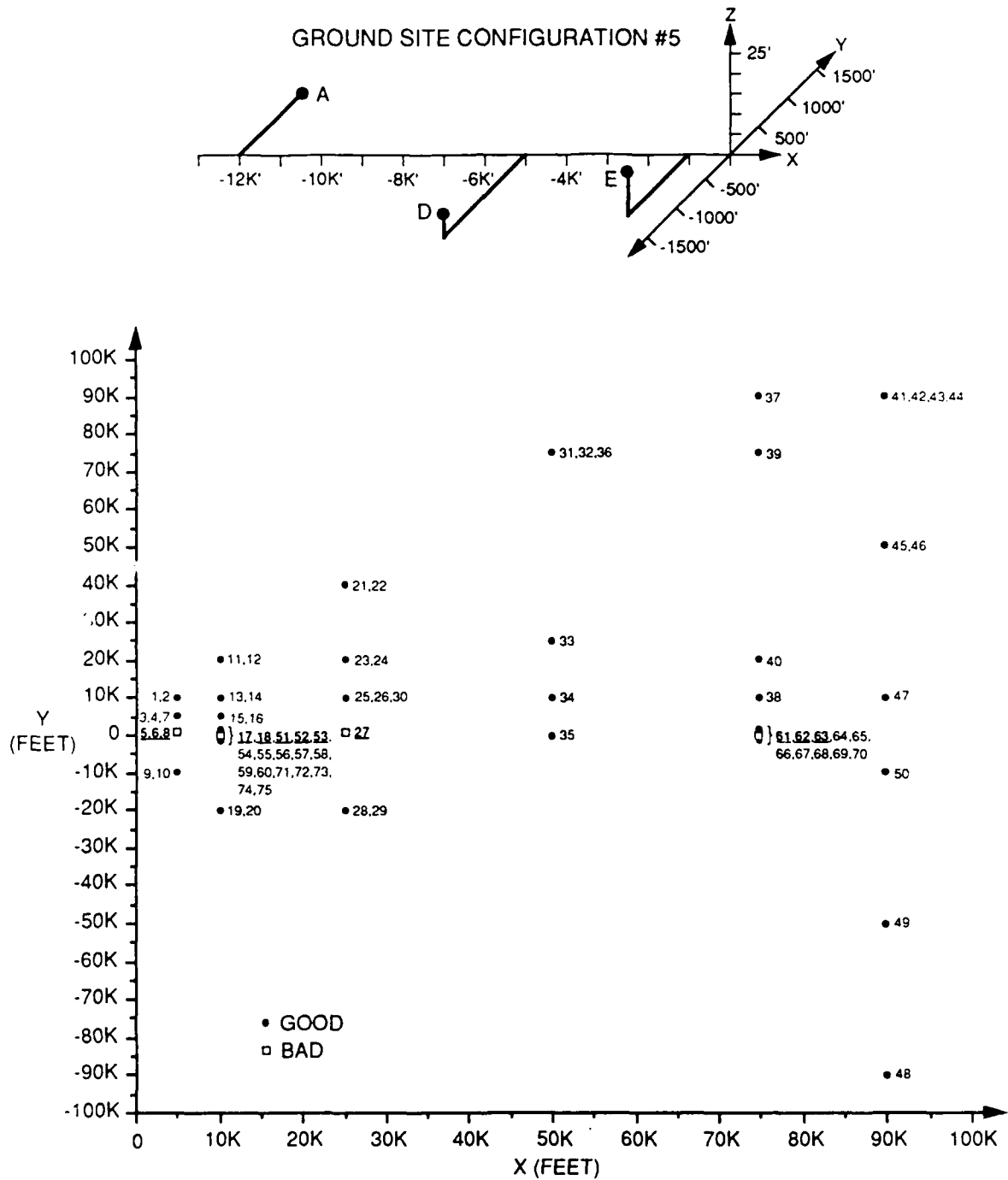
Figure 5-5. Performance Regions for Case 12 Simplified,
Ground Unit Siting 5

One attractive alternate may be taken from (4-2)-(4-4). Gather all terms on one side of the equation, and equate these to f, g, and h, respectively, consistent with the approach of [1]; this yields

$$f = x - x_D - [\rho^2 - (y - y_D)^2 - (z-z_D)^2]^{1/2} \qquad (5-6)$$
$$g = y - y_A + [(x - x_A)^2 + (z - z_A)^2]^{1/2}\tan\theta \qquad (5-7)$$
$$h = z - z_E - [(x - x_E)^2 + (y - y_E)^2]^{1/2}\tan\phi. \qquad (5-8)$$

This set has the Jacobian matrix

$$J = \begin{pmatrix} 1 & (y - y_D)/R_D & (z-z_D)/R_D \\ (x - x_A)\tan\theta/R_A & 1 & (z - z_A)\tan\theta/R_A \\ -(x - x_E)\tan\phi/R_E & -(y - y_E)\tan\phi/R_E & 1 \end{pmatrix} \qquad (5-9)$$

where

$$R_D = [\rho^2 - (y-y_D)^2 - (z - z_D)^2]^{1/2} \qquad (5-10)$$
$$R_A = [(x-x_A)^2 + (z - z_A)^2]^{1/2} \qquad (5-11)$$
$$R_E = [(x - x_E)^2 + (y-y_E)^2]^{1/2} \qquad (5-12)$$

This matrix is not known to have any singularities within coverage. For instance, if $\theta = \phi = 0$, the determinant reduces to 1. The determinant's dynamic range is very small, for the range of variables used above. But precautions must be taken lest the radicand in (5-10) be negative.

Instead of (5-6), it is possible to use (5-2), preferably multiplied by (1/2) to normalize. This produces a Jacobian matrix with the first row of $[(x - x_D)\ (y - y_D)\ (z - z_D)]$; if the ground units are collocated at the DME location there is a singularity at $x = x_D$, which is out of coverage. Otherwise, its properties are similar to those of the set (5-6)-(5-9). This will be used in the suggested alternate.

Another alternate uses

$$f = [(x - x_D)^2 + (y - y_D)^2 + (z-z_D)^2]^{1/2} - \rho \qquad (5-13)$$

instead of (5-6). The first row of its Jacobian is, similar to that described in the preceding paragraph, $[(x - x_D)/\rho\ (y - y_D)/\rho\ (z - z_D)/\rho]$. The properties are similar to those described above.

In the same spirit, multiply (5-7) and (5-8) by $\cos\theta$ and $\cos\phi$, respectively, and equate the results to new variables g and h. This approach has great similarity to the approach of [1], except that the trigonometric terms are not squared, there are thus only two solutions, of which only one is within coverage, and, as another consequence, no known singularities exist within coverage. This approach will be used to form the suggested alternate.

Several alternates for f have been described, and two have been suggested for g and h. Various systems may be formed by taking various combinations of the several f, g, and h forms. All share the common properties of a minimum number of possible solutions of which only one is normally within coverage, and no known singularities exist within coverage.

A specific choice for an alternate is now stated. As suggested above, it uses

$$f = (1/2)[(x-x_D)^2 + (y-y_D)^2 + (z-z_D)^2 - \rho^2] \qquad (5-14)$$
$$g = (y-y_A)\cos\theta + [(x-x_A)^2 + (z-z_A)^2]^{1/2}\sin\theta \qquad (5-15)$$
$$h = (z-z_E)\cos\phi - [(x-x_E)^2 + (y-y_E)^2]^{1/2}\sin\phi. \qquad (5-16)$$

Then the Jacobian partial derivatives matrix is

$$J = \begin{pmatrix} (x-x_D) & (y-y_D) & (z-z_D) \\ (x-x_A)\sin\theta/R_A & \cos\theta & (z-z_A)\sin\theta/R_A \\ -(x-x_E)\sin\phi/R_E & -(y-y_E)\sin\phi/R_E & \cos\phi \end{pmatrix} \qquad (5-17)$$

This algorithm can be generalized to include planar azimuth antenna patterns by multiplying $(z-z_A)$ by $(1-I56)$ in (5-15), in the radicand R_A in (5-11), and in the (2,3) location in J in (5-17).

This algorithm is compact, using 48 lines of code. It is believed to have the minimum number of solutions, so that the choice of initial conditions is relatively open. Any initialization with $x_0 > x_D$ and $x_T > x_D$ appears to converge to the correct solution. All known singularities are outside the coverage of the system.

It was shown above that the original and simplified forms of Case 12 have a problem near the azimuth antenna boresight; table 5-1 with case 3-70 was taken as the prototypical situation. In this same case, the performance of the alternate form, shown in table 5-2, is satisfactory, and exhibits no difficulties.

The dynamic range of the determinant of (5-17) is evaluated. Again assume the three ground units to be collocated at the origin. The determinant in this case is $\Delta = x\sec\theta\sec\phi$. With the same numerical assumptions, the dynamic range is 230. The dynamic range of the approach of [1] is greater by a factor of approximately 10^9. Code for this alternate algorithm is presented in appendix A.

5.5 COMPARISONS

Some of the properties of the three algorithms, Original Case 12, Simplified Case 12 and Alternate Case 12, are compared in table 5-3. The simplified and alternate versions are more compact and involve a lesser computation burden. The alternate is recommended, based on this table.

Table 5-2. Alternate Newton-Raphson Algorithm in Table 5-1 Case

```
                      GROUND STATION SITE GEOMETRY # 3
AZIMUTH ANTENNA SITE    DME TRANSMITTER SITE      ELEVATION ANTENNA SITE
  X      Y     Z          X       Y      Z          X       Y      Z
-6000. -1000.  10.      -1000.   500.    5.       -1000.   500.    5.


AIRCRAFT POSITION # 70.   OBSERVED DATA: RHO = 76061.1   THETA = .18   PHI = 1.88
   TRUE POS.    INIT. POS.
ITERATION NUMBER I    1          2          3          4          5          6
X 75000.00  76060.72  75013.98  75000.00  75000.00  75000.00  75000.00  75000.00
Y -1250.00   -234.64  -1250.04  -1250.00  -1250.00  -1250.00  -1250.00  -1250.00
Z  2500.00   2495.00   2500.23   2500.00   2500.00   2500.00   2500.00   2500.00
```

Table 5-3. Various Newton-Raphson Algorithms

	Original*	Simplified*	Alternate
Mean/Stand. Deviations of Iterations to Convergence	2.19/0.98	2.19/0.98	2.00/0.56
Lines of Code Out/In Iteration	10/99	40/9	37/11
No. of DO-loops	13	1	1
Operations (*,/) in Iterative Loop	94	87	65
Operations (sin, cos, $\sqrt{\ }$) in Iterative Loop	18	0	2
Dynamic Range	$2.3*10^{11}$	$2.3*10^{11}$	230
Singularities in Cat II Coverage	On Azimuth Centerline	On Azimuth Centerline	Between Azimuth & DME Sites

* Cases involving the singularity are disregarded.

SECTION 6

CONCLUSIONS

Of the various algorithms presented in [1] for position reconstruction in MLS area navigation operation, only two (Cases 9 and 12) cover general sitings of the ground units as well as conical-pattern azimuth antennas. These two can be generalized to handle planar antennas if that should be desired.

One of these algorithms, Case 9, using Gauss-Seidel iteration, exhibits slow convergence in some geometries at moderate azimuth angles, less than the minimum coverage of $\pm 40°$, and diverges at larger azimuth angles in most geometries. These problems may preclude its use in MLS RNAV operation. It should be noted that other Gauss-Seidel algorithms exist that are of similar size and complexity, but that do not diverge within the MLS coverage.

The other algorithm, Case 12, using Newton-Raphson iteration, converges rapidly everywhere except on or very near the centerline of the azimuth antenna. As formulated in [1] it consists of 109 lines of code and 13 DO-loops, and has the maximum number of possible false solutions; it thus requires careful choice of the essential initial conditions for the iterative procedure. Further, it has a singularity when the aircraft is on the azimuth antenna centerline (within the resolution due to quantization in the digital avionics), although an exact solution is possible; in this case, an unnecessary error signal is generated. Finally, for very small azimuth angles, near the azimuth antenna boresight, it is very near that singularity, and thus can yield a false solution. This formulation can be compacted, simplified and partially corrected. However, an alternate formulation is recommended as it uses 48 lines of code, has the minimum number of solutions and a much smaller dynamical range of numerical computation values, and imposes a smaller computational burden on the avionics. Further, it is believed to be without singularities within Category II coverage.

31

LIST OF REFERENCES

1. Radio Technical Commission for Aeronautics, 18 March 1988, "Minimum Operational Performance Standards for Airborne MLS Area Navigation Equipment," RTCA/DO-198, Appendix D, Washington, D.C.

2. United States Department of Transportation, Federal Aviation Agency, Program Engineering and Maintenance Service, October 1987, "Introduction to MLS," Washington, D.C.

3. International Civil Aeronautics Organization, October 1987, "International Standards, Recommended Practices and Procedures for Air Navigation Services; Aeronautical Communications, Annex 10," Fourth Edition of Volume I, April 1985, (Correction as of 22 October 1987), Montreal, Canada.

4. Radio Technical Commission for Aeronautics, July 1981, "Minimum Operational Performance Standards for Microwave Landing System (MLS) Airborne Receiving Equipment," RTCA/DO-177, Washington, D.C.

5. Strang, G., 1980, Linear Algebra and Its Applications, Second Edition, New York, NY: Academic Press, page 380.

APPENDIX A

ERRATA AND CODE

This appendix presents errata lists for the five general algorithms and the computer code where appropriate.

PLANAR ALGORITHMS

The planar algorithms are Cases 7, 10, and 11. A list of errata for these cases follows.

Case 7

On page 26 of [1], make the following changes:

Replace (7) by

$$"y^2(1+\cot^2\theta)+2y(x_D\cot\theta-x_A\cot\theta-y_A\cot^2\theta-y_D) \qquad (7)$$
$$+x_A{}^2+x_D{}^2+y_D{}^2+z_D{}^2+z^2-\rho^2+2(x_Ay_A\cot\theta-x_Dy_A\cot\theta-zz_D-x_Ax_D)+y_A{}^2\cot^2\theta=0"$$

This change corrects two errors in the analysis, repeated in the code, that cause the algorithm to converge to a wrong solution.

Replace (8) by

$$"y = [-B\pm(B^2-4AC)^{1/2}]/2A \qquad (8)"$$

The division by (2A) applies to the entire right-hand side of (8).

On page 29 of [1], make the following changes:

Replace (8) by

$$"y = [-B+(B^2-4AC)^{1/2}]/2A \text{ if } \theta < 0, \text{ and}$$
$$y = [-B-(B^2-4AC)^{1/2}]/2A \text{ if } \theta > 0. \qquad (8)"$$

The distinction between the positive and negative radical solutions is essential, otherwise the algorithm yields false results for $\theta > 0$. This error is repeated in the code in [1]. Also, the division by (2A) applies to the entire right-hand side of (8).

Corrected code is not provided, since the planar-azimuth case is now of reduced interest.

35

Case 10

Note that this case can be generalized to conical by changing (2), and the subsequent analysis, as shown in section 3.

On page 38 of [1], replace (5) by

$$"z = z_E + [(x-x_E)^2+(y-y_E)^2]^{1/2}\tan\phi \qquad (5)"$$

and replace (9) by

$$"z = z_E + [(x-x_E)^2+(y-y_E)^2]^{1/2}\tan\phi \qquad (9)"$$

to define correctly the range of application of the square-root operation and the coefficient of $\tan\phi$.

On page 39, replace (12) by

$$"z_{i+1} = z_E + [(x_{i+1}-x_E)^2+(y_{i+1}-y_E)^2]^{1/2}\tan\phi \qquad (12)"$$

This change ensures that the correct sign of $\tan\phi$ is selected and that the coefficient of $\tan\phi$ is correctly defined.

Corrected code is not presented.

Case 11

On page 43 of [1], as (2A) divides the entire right of (8), make the following change:

$"x = [-B+(B^2-4AC)^{1/2}]/2A$ if $x > x_D$, since $x > x_D$ is the desired solution. $\qquad (8)"$

On page 44 of [1], make the following change:

From the paragraph below (13), "Please note that to avoid singularities, the angle ...axis.", delete the underlined words, as this method has no singularities; the definition of the positive direction of the angle θ repeats the definition established elsewhere in [1] and [5].

Corrected code is not provided.

CONICAL ALGORITHMS

Two conical azimuth antenna algorithms (Cases 9 and 12) are presented in [1]. Errata and alternate code are given below where appropriate.

On page 34, correct the signs in (5) to read

$$"y = y_A - [(x-x_A)^2 + (z-z_A)^2]^{1/2} \tan\theta \qquad (5)"$$

On page 34, provide subscripts for (9), as

$$"z_{i+1} = z_E + [(x_i-x_E)^2 + (y_i-y_E)^2]^{1/2} \tan\phi \qquad (9)"$$

Correct the signs in (10), and provide subscripts, to read

$$"y_{i+1} = y_A - [(x_i-x_A)^2 + (z_{i+1}-z_A)^2]^{1/2} \tan\theta \qquad (10)"$$

If desired, multiply $(z-z_A)$ in (5) and (10) by (1-I56) to generalize to include planar azimuth.

On page 35, near the bottom, correct the computer code as follows:

Identify the three lines below "IFLAG=0..." as comments. The code cannot compile as given.

To the "NOTE", add to the comment, "For some geometries, this algorithm diverges for azimuth angles less than 40°."

On page 36, near the bottom of the code, after

"IF(ABS(Z1-Z).GT.LIMZ)GO TO 20"

insert the following lines

```
"Z=Z1
 Y=Y1
 X=X1
 GOTO 99"
```

Otherwise, the algorithm always iterates 10 times and generates the flag for excessive iterations.

These changes are so few and simple that it is not necessary to present the entire corrected code.

Case 12

The errors in the analysis are noted; they do not appea. on the code.

On page 49, in the first line of the square matrix in (22) and (23):

Replace "fg" by "fz".

On page 50, replace the initial conditions in (29) by

$$
\begin{aligned}
"x_0 &= \rho\cos\theta \\
y_0 &= -\rho\sin\theta \\
z_0 &= \rho\sin\phi"
\end{aligned}
\tag{29}
$$

as the initial conditions for y_0 and z_0 are incorrect.

The import of the last paragraph is clarified by inserting the underlined words in the first sentence, so that it reads,

"...azimuth centerline ($\theta=0$), assumed parallel to the runway centerline, which...".

The code can be compressed and the special case of singularity of the Jacobian matrix when $\theta = 0$ can be correctly handled in the code presented below, from [2], as figure A-1. However, this does not necessarily fully treat the situation when the iteration (or aircraft) is near, but not exactly on, the singularity condition.

The code for the alternate formulation of this approach, discussed in section 5, above, is presented in figure A-2. It was noted above that this approach has a singularity at $x = x_D$ if the ground units are collocated. However, there is no singularity at or near $y = y_A$ nor $z = z_E$ in the collocated case. It is possible that there is no singularity within coverage; however, this has not been proved.

```
                SUBROUTINE CASE12S(XD,YD,ZD,XA,YA,ZA,XE,YE,ZE,RHO
           1    ,THETA,PHI,XT,YT,ZT,ARAOUT,IFLAG,ITER)
                DIMENSION ARAOUT(3,10)
C               DEFINE TRIG FUNCTIONS OF OBSERVED ANGLES.
                CTHETA=COS(THETA)
                STHETA=SIN(THETA)
                SPHI=SIN(PHI)
                CPHI=COS(PHI)
                ARAOUT(1,1)=XT
                ARAOUT(2,1)=YT
                ARAOUT(3,1)=ZT
C               INITIALIZATION.
                XHAT=RHO*CTHETA
                YHAT=-RHO*STHETA
                ZHAT=RHO*SPHI
C               STORE INITIALIZATIONS IN OUTPUT ARRAY, ARAOUT.
                ARAOUT(1.2)=XHAT
                ARAOUT(2,2)=YHAT
                ARAOUT(3,2)=ZHAT
C               END OF INITIALIZATIONS.
C               THE SIMPLIFIED ITERATIVE RECONSTRUCTION ALGORITHM BEGINS HERE.
                DO 10 I=1,ITER
C               FORM VECTOR OF FUNCTIONS.
                F1=(XHAT-XD)**2+(YHAT-YD)**2+(ZHAT-ZD)**2-RHO**2
                F2=-(CTHETA*(YHAT-YA))**2
           1    +(STHETA*(XHAT-XA))**2+(STHETA*(ZHAT-ZA))**2
                F3=(SPHI*(XHAT-XE))**2+(SPHI*(YHAT-YE))**2
           1    -(CPHI*(ZHAT-ZE))**2
C               FORM JACOBIAN MATRIX ELEMENTS
                C11=2.0*(XHAT-XD)
                C12=2.0*(YHAT-YD)
                C13=2.0*(ZHAT-ZD)
                C21=2.0*(STHETA*STHETA)*(XHAT-XA)
                C22=-2.0*(CTHETA*CTHETA)*(YHAT-YA)
                C23=2.0*(STHETA*STHETA)*(ZHAT-ZA)
                C31=2.0*(SPHI*SPHI)*(XHAT-XE)
                C32=2.0*(SPHI*SPHI)*(YHAT-YE)
                C33=-2.0*(CPHI*CPHI)*(ZHAT-ZE)
C               TREATMENT OF SINGULARITY AT Y=YA OR AT THETA=0.
                IF((THETA.NE.0.0).OR.((YHAT-YA).NE.0.0))GOTO 11
                DET2=C11*C33-C13*C31
C               NOTE THAT DET2 > 0.
                C2IN11=C33/DET2
                C2IN12=-C13/DET2
                C2IN21=-C31/DET2
                C2IN22=C11/DET2
```

Figure A-1. Code for Case 12 Simplified

```
              DELTX=-(C2IN11*F1+C2IN12*F3)
              DELTY=0.0
              DELTZ=-(C2IN21*F1+C2IN22*F3)
              XHAT=XHAT+DELTX
              YHAT=YA
              ZHAT=ZHAT+DELTZ
              IFLAG=2
              GOTO 12
C             MATRIX INVERSION, SIMPLIFIED, BUT UNIQUE TO A 3X3.
11            CONTINUE
              DET3=C11*C22*C33+C21*C32*C13+C31*C12*C23
     1        -C13*C22*C31-C23*C32*C11-C33*C21*C12
C             TEST 3X3 MATRIX FOR SINGULARITY.
              IF(DET3.EQ.0.0)GOTO 13
              CINV11=(C22*C33-C23*C32)
              CINV12=(C13*C32-C12*C33)
              CINV13=(C12*C23-C13*C22)
              CINV21=(C23*C31-C21*C33)
              CINV22=(C11*C33-C13*C31)
              CINV23=(C13*C21-C11*C23)
              CINV31=(C21*C32-C22*C31)
              CINV32=(C12*C31-C11*C32)
              CINV33=(C11*C22-C12*C21)
              IFLAG=1
C             END OF INVERSION.
C             PRODUCT OF MATRIX-INVERSE AND VECTOR.
              DELTX=-(CINV11*F1+CINV12*F2+CINV13*F3)/DET3
              DELTY=-(CINV21*F1+CINV22*F2+CINV23*F3)/DET3
              DELTZ=-(CINV31*F1+CINV32*F2+CINV33*F3)/DET3
C             NOW CREATE NEW XHAT, YHAT, & ZHAT.
              XHAT=XHAT+DELTX
              YHAT=YHAT+DELTY
              ZHAT=ZHAT+DELTZ
              GOTO 12
C             TRAP AND MARKER FOR SINGULAR MATRIX.
13            IFLAG=3
              XHAT=0.0
              YHAT=0.0
              ZHAT=1000.0
12            CONTINUE
              ARAOUT(1,(2+I))=XHAT
              ARAOUT(2,(2+I))=YHAT
              ARAOUT(3,(2+I))=ZHAT
10            CONTINUE
              RETURN
              END
C             END OF SIMULATION SUBROUTINE.
```

Figure A-1. Code for Case 12 Simplified (concluded)

```fortran
      SUBROUTINE CASE12A(XD,YD,ZD,XA,YA,ZA,X1,YE,ZE,RHO,THETA,PHI
     1 ,XT,YT,ZT,ARAOUT,IFLAG,ITER,I56)
      DIMENSION ARAOUT(3,10)
C     ALTERNATE NEWTON-RAPHSON ALGORITHM
C     INITIALIZE ANGLE DATA
      STHETA=SIN(THETA)
      CTHETA=COS(THETA)
      SPHI=SIN(PHI)
      CPHI=COS(PHI)
c     STORE TRUE VALUES IN OUTPUT ARRAY
      ARAOUT(1,1)=XT
      ARAOUT(2,1)=YT
      ARAOUT(3,1)=ZT
C     INITIALIZATION OF POSITION ESTIMATE
      XHAT=RHO*CTHETA
      YHAT=-RHO*STHETA
      ZHAT=RHO*SPHI
C     STORE INITIALIZATIONS IN OUTPUT ARRAY
      ARAOUT(1,2)=XHAT
      ARAOUT(2,2)=YHAT
      ARAOUT(3,2)=ZHAT
C     END OF INITIALIZATIONS
C     THE ITERATIVE RECONSTRUCTION ALGORITHM BEGINS HERE
      DO 10 I=1,ITER
C     SEE (5-14)-(5-16)
      F1=((XHAT-XD)**2+(YHAT-YD)**2+(ZHAT-ZD)**2-RHO**2)/2.0
      RA=SQRT((XHAT-XA)**2+(1-I56)*(ZHAT-ZA)**2)
      F2=(YHAT-YA)*CTHETA+RA*STHETA
      RE=SQRT((XHAT-XE)**2+(YHAT-YE)**2)
      F3=(ZHAT-ZE)*CPHI-RE*SPHI
C     MATRIX ELEMENTS; SEE (5-17)
      C11=(XHAT-XD)
      C12=(YHAT-YD)
      C13=(ZHAT-ZD)
      C21=(XHAT-XA)*STHETA/RA
      C22=CTHETA
      C23=(1-I56)*(ZHAT-ZA)*STHETA/RA
      C31=-(XHAT-XE)*SPHI/RE
      C32=-(YHAT-YE)*SPHI/RE
      C33=CPHI
```

Figure A-2. Code for Case 12 Alternate

41

```
C          MATRIX INVERT, UNIQUE TO A 3X3
11         CONTINUE
           DET3=C11*C22*C33+C21*C32*C13+C31*C12*C23
     1     -C13*C22*C31-C23*C32*C11-C33*C21*C12
C          TREATMENT OF THE SINGULARITY, IF ANY
           IF(DET3 .LE. 1.0)GOTO 13
           CINV11=(C22*C33-C23*C32)
           CINV12=(C13*C32-C12*C33)
           CINV13=(C12*C23-C13*C22)
           CINV21=(C23*C31-C21*C33)
           CINV22=(C11*C33-C13*C31)
           CINV23=(C13*C21-C11*C23)
           CINV31=(C21*C32-C22*C31)
           CINV32=(C12*C31-C11*C32)
           CINV33=(C11*C22-C12*C21)
           IFLAG=1
C          END OF INVERSION
C          PRODUCT OF MATRIX-INVERSE AND VECTOR
           DELTX=-(CINV11*F1+CINV12*F2+CINV13*F3)/DET3
           DELTY=-(CINV21*F1+CINV22*F2+CINV23*F3)/DET3
           DELTZ=-(CINV31*F1+CINV32*F2+CINV33*F3)/DET3
C          NOW CREATE NEW XHAT, ETC.
           XHAT=XHAT+DELTX
           YHAT=YHAT+DELTY
           ZHAT=ZHAT+DELTZ
           GOTO 12
13         IFLAG=3
           XHAT=0.0
           YHAT=0.0
           ZHAT=1000.0
12         CONTINUE
C          END OF SIMULATION
C          OUTPUT ARRAY FOLLOWS
           ARAOUT(1,(2+I))=XHAT
           ARAOUT(2,(2+I))=YHAT
           ARAOUT(3,(2+I))=ZHAT
C          THIS COMPLETES THE ITERATION
10         CONTINUE
```

Figure A-2. Code for Case 12 Alternate (concluded)

APPENDIX B

DATABASE

The databases used to exercise the several algorithms are presented in this appendix. Table B-1 presents the locations of the five arrangements of the ground units, and table B-2 presents the 50 aircraft locations common to both the Case 9 and Case 12 exercises. Table B-3 presents the set of 25 additional aircraft locations used to delineate the areas in which Case 9 has problems, and table B-4 presents the set of 25 additional locations used to explore the region where Case 12 has problems. Plan views of these data appeared in the several figures in section 2.

Table B-1. Ground Unit Locations

Set #	DME			Azimuth Antenna			Elevation Antenna		
	x_D	y_D	z_D	x_A	y_A	z_A	x_E	y_E	z_E
1	-1000	500	5	-1000	500	5	-1000	500	5
2	-6000	0	5	-6000	0	5	-1000	500	5
3	-1000	500	5	-6000	-1000	10	-1000	500	5
4	-5000	-1000	25	-10000	-500	5	-1000	500	5
5	-5000	-1500	5	-12000	1000	0	-1000	-1000	10

Table B-2. Aircraft Locations for Common Database

Location Number	Aircraft Location X_T	Y_T	Z_T	Location Number	Aircraft Location X_T	Y_T	Z_T
1	5000	10000	3000	26	25000	10000	2500
2	5000	10000	1000	27	25000	1000	2500
3	5000	5000	3000	28	25000	-20000	7500
4	5000	5000	1000	29	25000	-20000	2500
5	5000	1000	3000	30	25000	10000	5000
6	5000	1000	1000	31	50000	75000	10000
7	5000	5000	500	32	50000	75000	2500
8	5000	1000	500	33	50000	25000	10000
9	5000	-10000	3500	34	50000	10000	5000
10	5000	-10000	1000	35	50000	0	2500
11	10000	20000	4000	36	50000	75000	25000
12	10000	20000	1000	37	75000	90000	10000
13	10000	10000	4000	38	75000	10000	5000
14	10000	10000	1000	39	75000	75000	10000
15	10000	5000	4000	40	75000	20000	1000
16	10000	5000	1000	41	90000	90000	20000
17	10000	1000	4000	42	90000	90000	25000
18	10000	1000	1000	43	90000	90000	10000
19	10000	-20000	4000	44	90000	90000	5000
20	10000	-20000	1000	45	90000	50000	20000
21	25000	40000	5000	46	90000	50000	2000
22	25000	40000	1000	47	90000	10000	10000
23	25000	20000	5000	48	90000	-90000	15000
24	25000	20000	1000	49	90000	-50000	10000
25	25000	10000	7500	50	90000	-10000	20000

Table B-3. Added Aircraft Locations for Examination of Case 9

Location Number	Aircraft Location			Location Number	Aircraft Location		
	X_T	Y_T	Z_T		X_T	Y_T	Z_T
51	10000	18000	5000	64	25000	-35000	5000
52	10000	16000	5000	65	25000	-30000	5000
53	10000	14000	5000	66	25000	-25000	5000
54	10000	12000	5000	67	50000	70000	5000
55	10000	8000	5000	68	50000	60000	5000
56	10000	-18000	5000	69	50000	50000	5000
57	10000	-16000	5000	70	50000	40000	5000
58	10000	-14000	5000	71	50000	-70000	5000
59	10000	-12000	5000	72	90000	40000	5000
60	10000	-8000	5000	73	90000	30000	5000
61	25000	35000	5000	74	90000	20000	5000
62	25000	30000	5000	75	90000	40000	20000
63	25000	25000	5000				

Table B-4. Added Aircraft Locations for Examination of Case 12

Location Number	Aircraft Location			Location Number	Aircraft Location		
	X_T	Y_T	Z_T		X_T	Y_T	Z_T
51	10000	1500	1000	64	75000	750	2500
52	10000	1250	1000	65	7500	500	2500
53	10000	1000	1000	66	75000	250	2500
54	10000	750	1000	67	75000	0	2500
55	10000	500	1000	68	75000	-250	2500
56	10000	250	1000	69	75000	-750	2500
57	10000	0	1000	70	75000	-1250	2500
58	10000	-250	1000	71	10000	25	1000
59	10000	-750	1000	72	10000	50	1000
60	10000	-1250	1000	73	10000	100	1000
61	75000	1500	2500	74	10000	150	1000
62	75000	1250	2500	75	10000	200	1000
63	75000	1000	2500				

APPENDIX C

INITIALIZATION

Gauss-Seidel and Newton-Raphson iterative algorithms must be initialized to establish the start for the search procedure. If the mathematics permit multiple solutions, the selection of the initial condition determines which of the solutions is reached. A variety of satisfactory initial conditions are presented in [1], under the assumption that the aircraft true position has $x_T > x_D$. If this assumption is false, then multiple solutions may be within coverage; the problem of selecting the correct initial condition in this case has not been solved.
The initial condition procedures are presented below in the order of increasing complexity. Note that the algorithm of Case 9 does not use z_0; however, this is not a universal property of Gauss-Seidel algorithms, but only of the specific algorithm offered in [1].

Simplest initialization

$$x_0 = \rho\cos\theta$$
$$y_0 = \rho\sin\theta$$
$$z_0 = \rho\sin\phi$$

Intermediate complexity initialization

$$x_0 = x_D + \rho\cos\theta$$
$$y_0 = x_A - \rho\sin\theta$$
$$z_0 = z_E + \rho\sin\phi$$

Higher complexity initialization

$$x_0 = x_D + \rho\cos\theta\cos\phi$$
$$y_0 = x_A - \rho\sin\theta\cos\phi$$
$$z_0 = z_E + \rho\sin\phi$$

Other initializations, using more information about the geometry, may also be devised. However, those above span the usual range of variety.

In geometries where an algorithm converges slowly, a very good initial condition may save one, or perhaps even two iterations. However, a very good initial condition also involves more complexity, and the time used in evaluating the more complex expression may become significant.